

# AVR101 AVR<sup>®</sup> MCU 的基礎架構 利用 MCC Melody 的設定、開發方式 以及 I/O, Timer, UART, 中斷操作

---

# 課程目標

胸有成竹



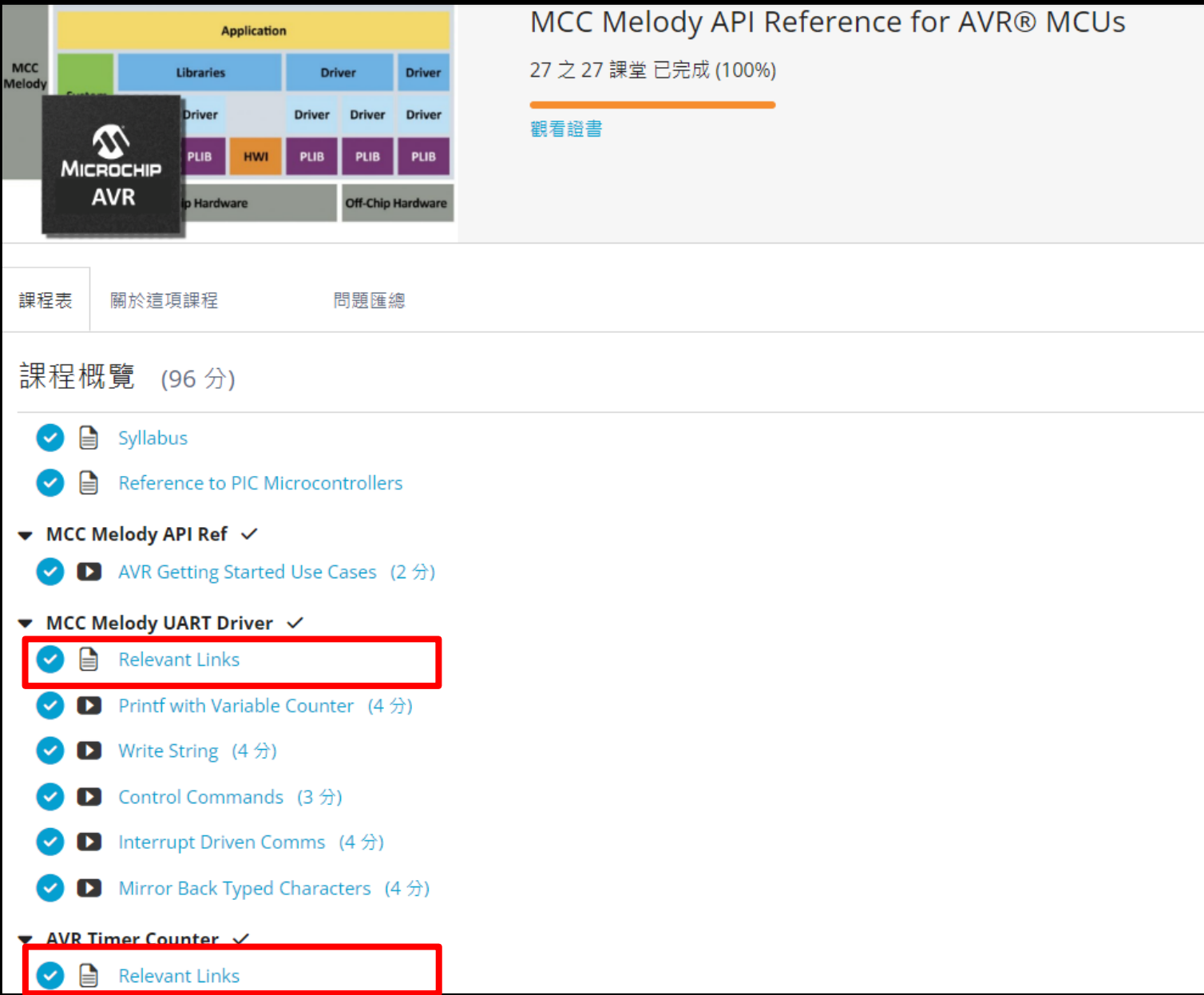
見招拆招

# AVR-101 課程介紹

本課程為 **Microchip AVR MCU** 系列的入門課程，綜合了以下 **3 個 Microchip MU** 課程，將課程的內容歸納整理後讓參加課程的學員能將線上課程所學的內容實際地應用於 **AVR MCU** 的開發工作

 <b>MPLAB® Code Configurator (MCC)</b> <b>For Simplified Embedded Software Development</b>  The MPLAB® Code Configurator (MCC) is a free plug-in of MPLAB® X IDE, which provides an easy setup and configuration experience for supported microcontrollers. (Dec 2020)  免費 75 分	 <b>Overview of the Microchip Code Configurator (MCC) Content Manager (CM)</b>  This course provides basic information on how to navigate and create a new project using the MCC Content Manager. (June 2022)  免費 13 分	 <b>MCC Melody API Reference for AVR® MCUs</b>  MCC Melody components are the collective name for the provided libraries, drivers, Peripheral Libraries (PLIBs) and Hardware Initializers (HWIs). For select components, the MCC Melody API references now include a "How to use the ..." sections, which are a series of use cases. (Jan 2023)  免費 96 分
---	--	--

# MU 的課程內容也含了必要的參考連結



MCC Melody API Reference for AVR® MCUs

27 之 27 課堂 已完成 (100%)

[觀看證書](#)

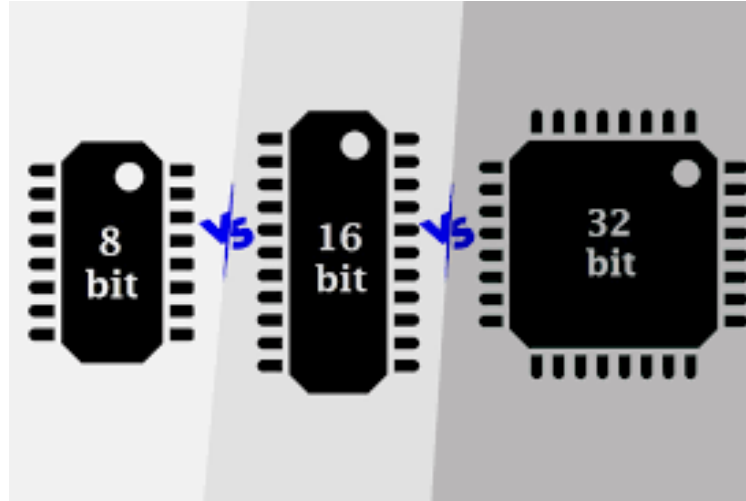
課程表 關於這項課程 問題匯總

課程概覽 (96 分)

- ✓ [Syllabus](#)
- ✓ [Reference to PIC Microcontrollers](#)
- ▼ **MCC Melody API Ref** ✓
  - ✓ [AVR Getting Started Use Cases](#) (2 分)
- ▼ **MCC Melody UART Driver** ✓
  - ✓ [Relevant Links](#)
  - ✓ [Printf with Variable Counter](#) (4 分)
  - ✓ [Write String](#) (4 分)
  - ✓ [Control Commands](#) (3 分)
  - ✓ [Interrupt Driven Comms](#) (4 分)
  - ✓ [Mirror Back Typed Characters](#) (4 分)
- ▼ **AVR Timer Counter** ✓
  - ✓ [Relevant Links](#)



# Which MCU is most suitable for my Project ??



# 為何選擇 Microchip AVR® MCU and PIC®MCU?

- 適合所有通用的創新 / Microchip持續地對特性和功能進行實質投資
  - 可自行客製的內建周邊以適應多種應用
  - 在所有的前沿應用幾乎都可看到 8-bit MCU 的身影
- 具備所有使用者需要的彈性，涵蓋以下所列的各項目
  - Portfolio Flexibility
  - System Flexibility
  - Chip-Level Flexibility
  - Purchase Flexibility
- 專注於您完整的設計之的過程
  - 直觀並且易用的開發工具

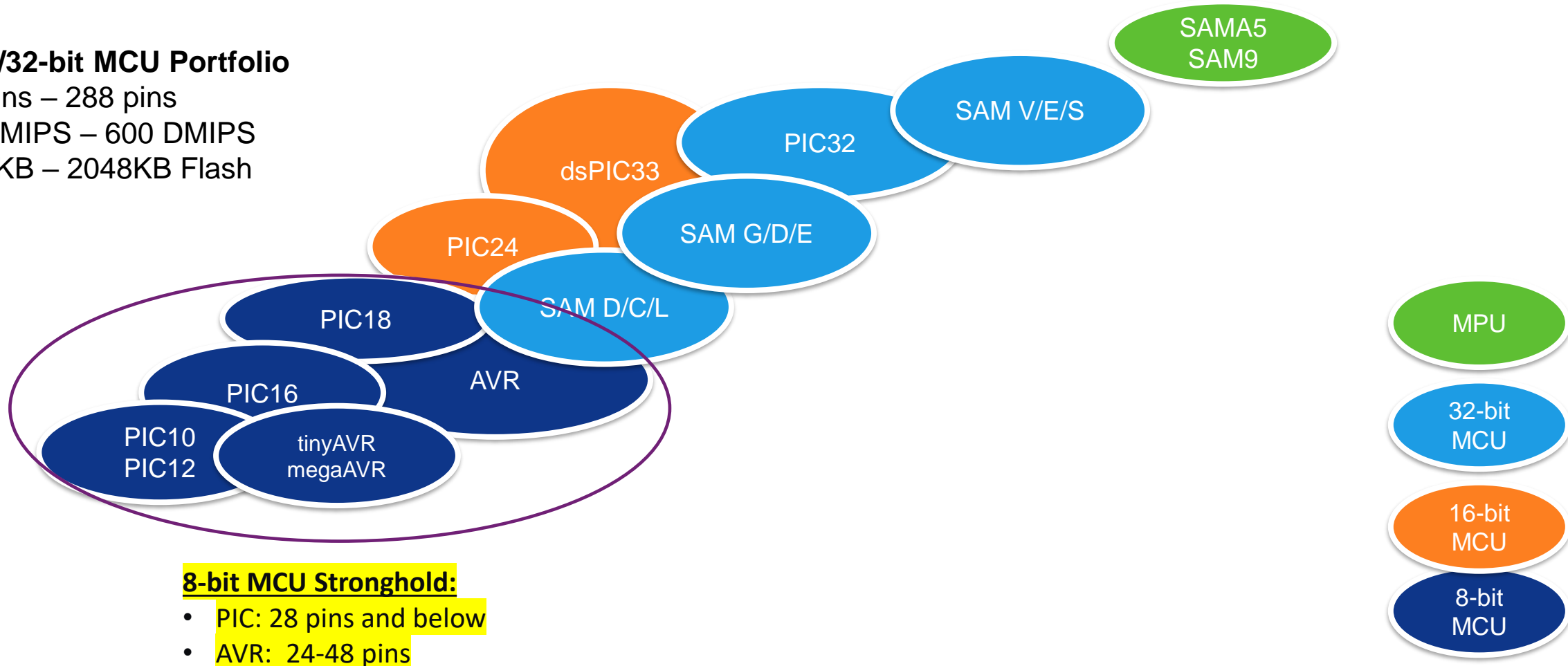
簡單、易用、功能創新以及不斷推出增強的新型號

# Microchip MCUs : A Product for Every Need

2000+ Products – Broadest Portfolio – Unified Comprehensive Ecosystem

## 8-/16-/32-bit MCU Portfolio

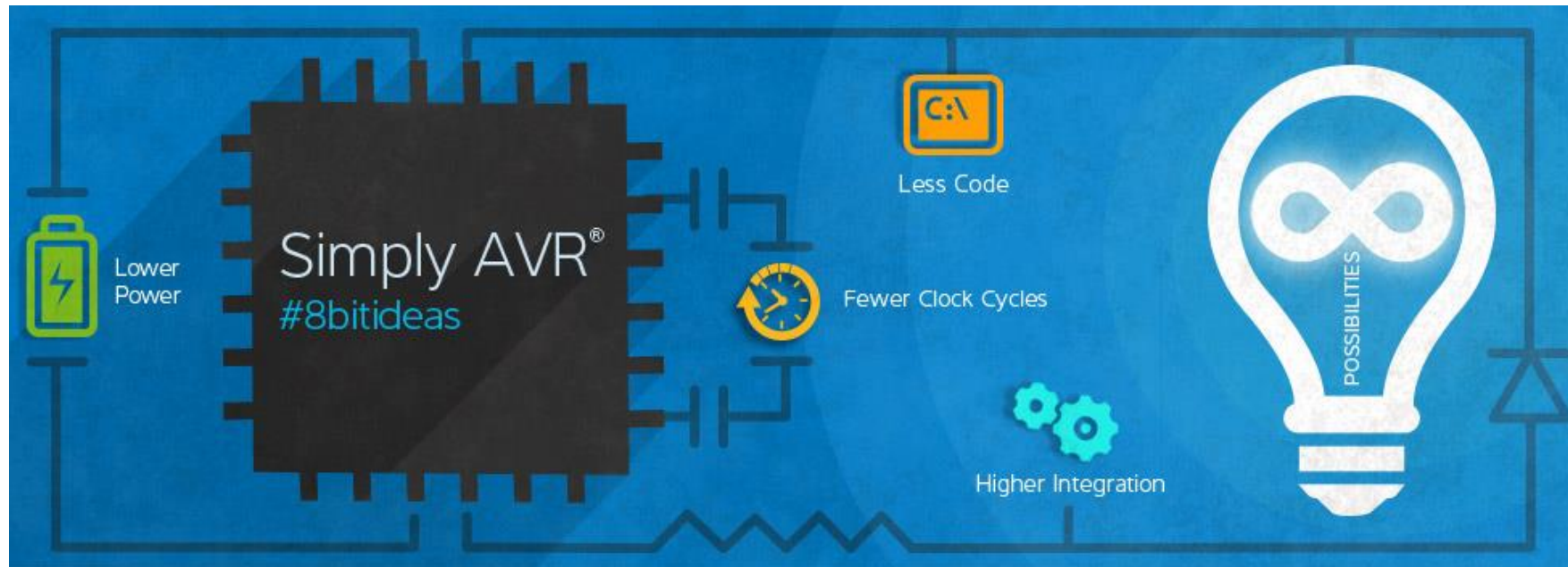
- 6 pins – 288 pins
- 8 DMIPS – 600 DMIPS
- 0.5KB – 2048KB Flash



### 8-bit MCU Stronghold:

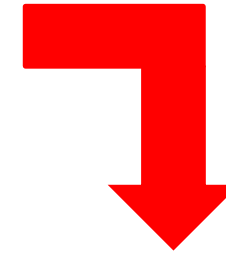
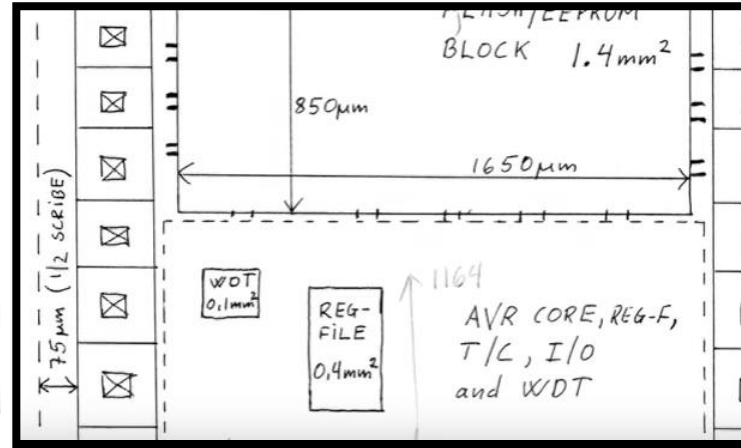
- PIC: 28 pins and below
- AVR: 24-48 pins

# AVR<sup>®</sup> Microcontrollers 簡介

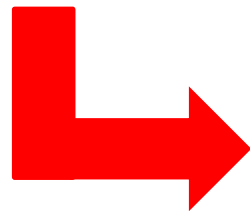
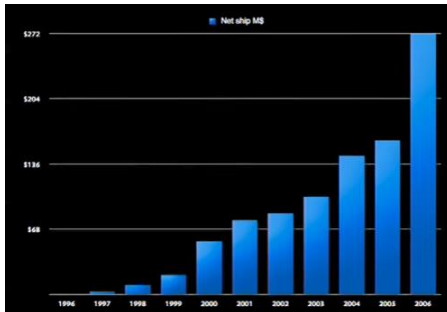
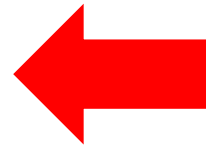


# AVR<sup>®</sup> Microcontrollers – 架構起源於挪威理工學院

1992



1997



2016



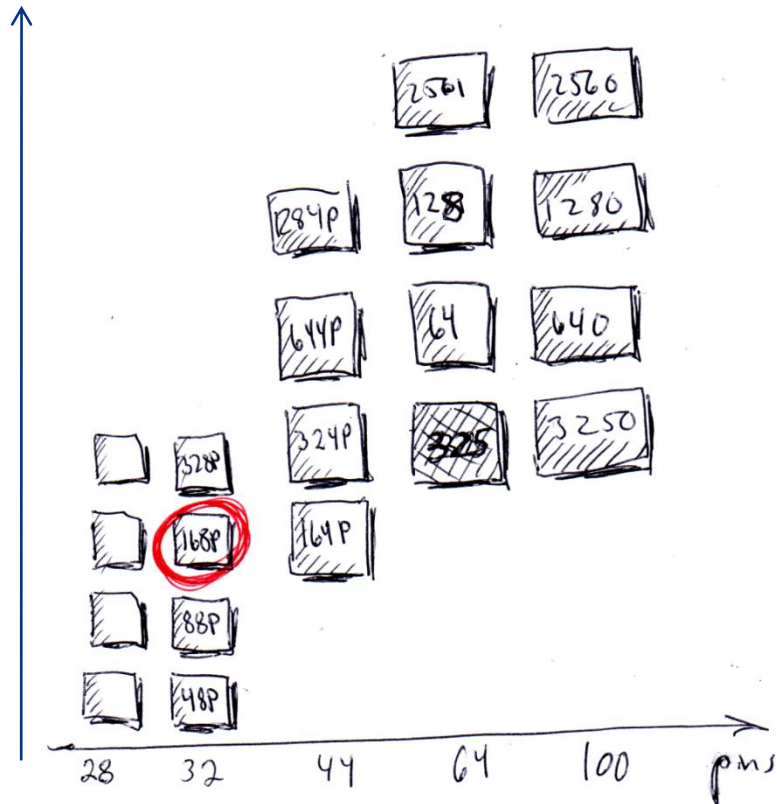
# MICROCHIP



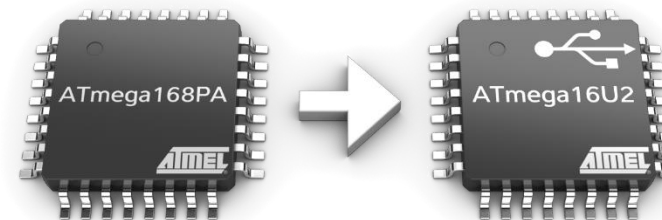


# AVR<sup>®</sup> MCU 的設計目標：滿足設計上的所有需求

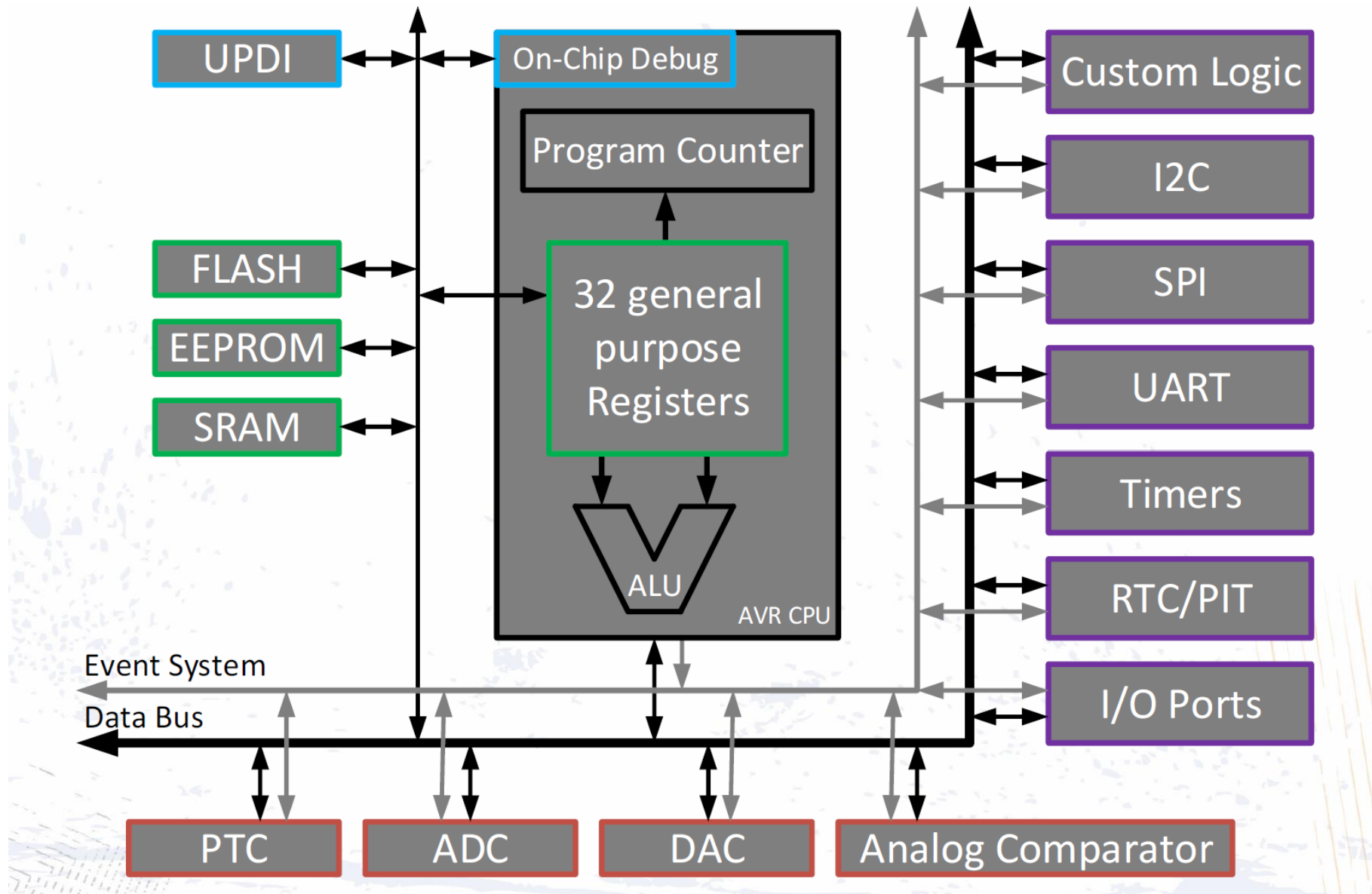
- 0.5– 384KB Flash
- 6– 100 pins
- Common peripherals



- EEPROM
- Flash size ~32K (may need USB, then what?????)
- Capacitive touch interface
- I2C
- UART or SPI
- 3 yr min battery life
- LEDs
- Bootloader
- USB to be added next year
- Reuse code from project charlie
- Customer wants to see prototype yesterday... argh!!!

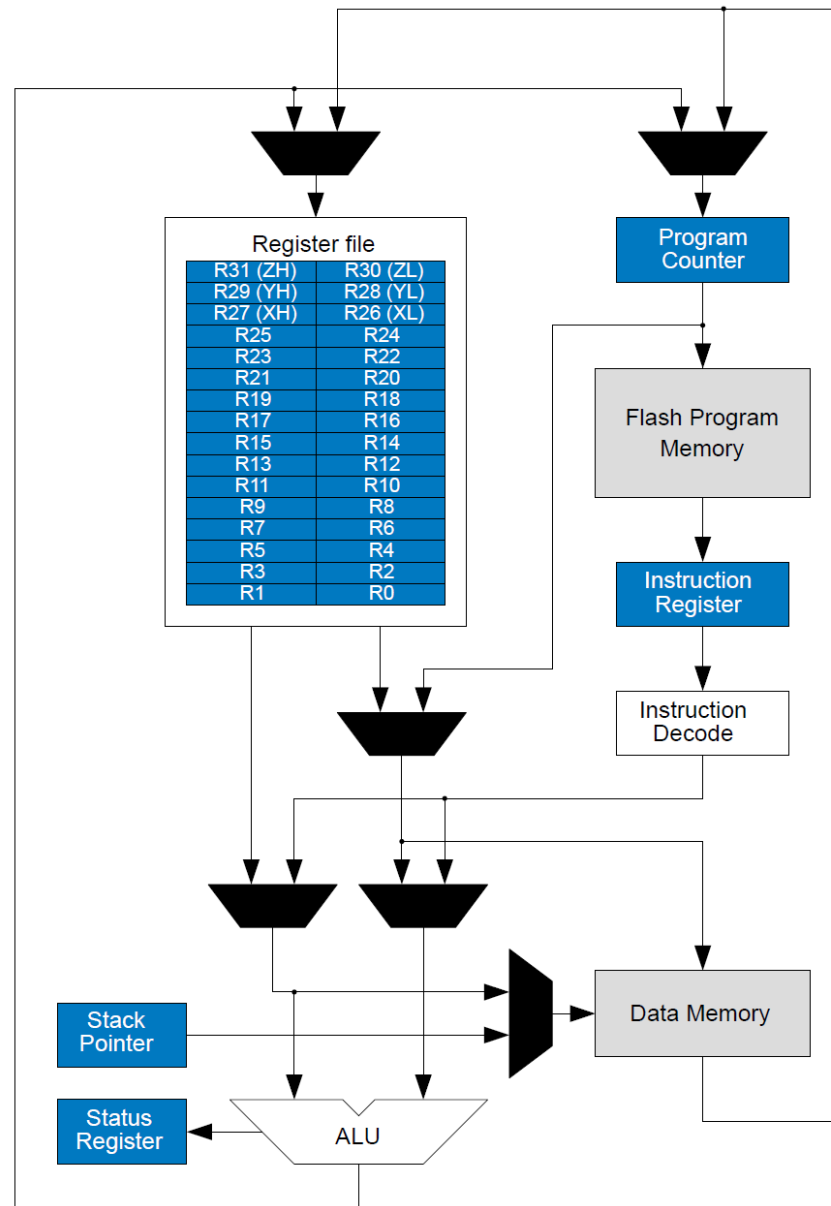


# AVR<sup>®</sup> MCU 的通用架構





# AVR<sup>®</sup> CPU Architecture

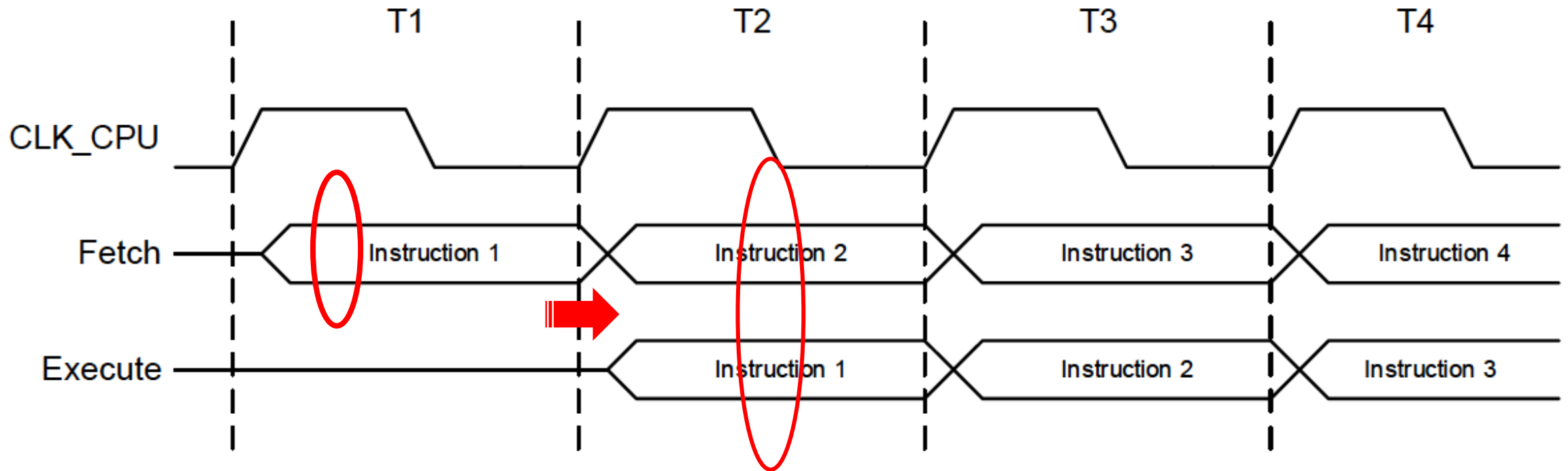


# AVR® CPU 的 Arithmetic Logic Unit (ALU)

- **算術邏輯單元 (ALU)** 支持工作暫存器之間或暫存器之間的算術和邏輯運算
  - 也支援常數和工作暫存器運算
  - 還可以執行單一暫存器操作
- **ALU 與暫存器文件中的所有 32 個通用工作暫存器直接連接運行**
  - 工作暫存器之間或工作暫存器與立即數操作數之間的算術運算是在一個時脈週期內執行
  - 經過算術或邏輯運算後，結果存儲在暫存器文件(Register File )中
- **狀態暫存器 (CPU.SREG)** 在運算後會被更新以反映有關操作結果的信息
- **ALU 運算分為三大類：算術運算、邏輯運算和位元運算**
- 支援 8- 以及 16-bit 算術，指令集也允許高效實現32位算術
- 硬體乘法器支持有號和無號乘法以及小數格式

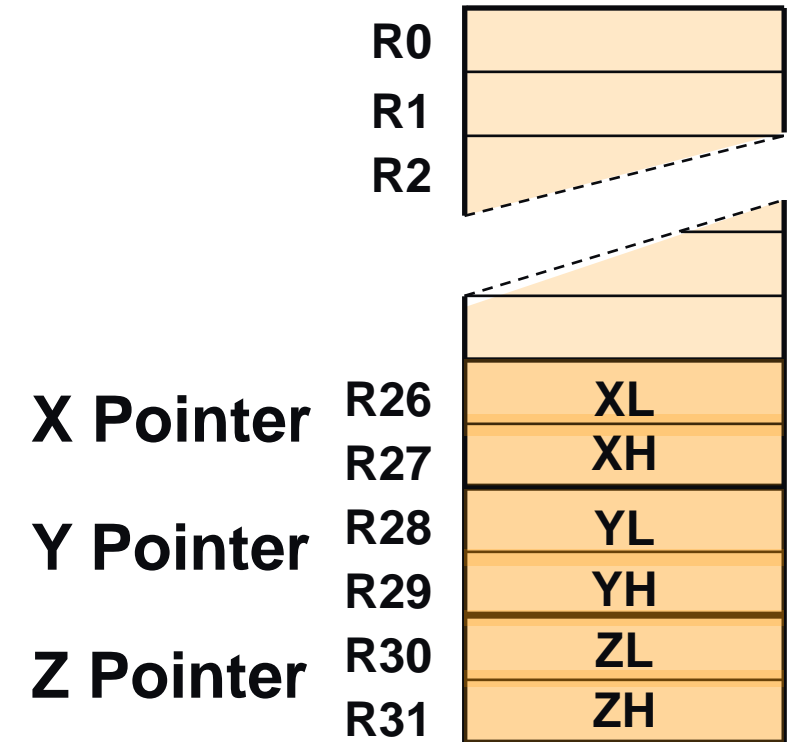
# AVR® CPU 的 Single Level Pipeline

除第一個指令外，**連續性程式**的執行與下一個指令的提取可以在同一個時脈週期完成



# 32 General Purpose Registers

- **AVR 總共有 32 個 8-bit 通用暫存器**
  - ✓ 相當於有大量的 Accumulator
  - ✓ 都可以被 ALU 直接存取
  - ✓ 如此可盡量減少資料的搬移次數
- **3 register pairs act as 16-bit data pointers**
  - ✓ Instruction for auto increment, decrement and displacement
  - ✓ Enabling efficient address calculations
- **Separate Program Stack Pointer**
  - ✓ Separate stacks for return addresses and local variables
- **Z Pointer suited for look up tables and indirect jump**
- **Direct address of up to 64K**
  - ✓ With optional extension to 20-bit (2M)

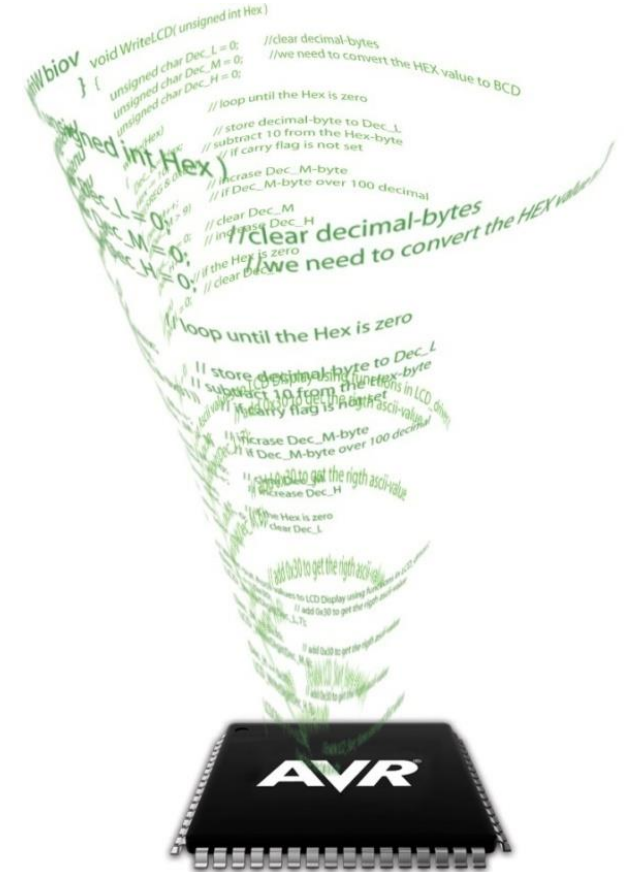


# AVR<sup>®</sup> MCU Architecture – 極佳的 Code Density

Compact code leaves more room for data storage, lowers cost

- Architecture designed for C language
- 32 general registers
- **C-like addressing modes**
- 16- and 32-bit arithmetic support
- Variables on the C stack can be manipulated
- Linear address maps

Example of code sizes [bytes]			
Code	AVR	STM8	MSP430
8-bit Matrix	126	134	86
8-bit Math	56	67	178
8-bit Switch	102	168	198
<b>Total</b>	<b>284</b>	<b>369</b>	<b>462</b>



# AVR® & PIC® MCU都有著豐富的內建周邊

## Intelligent Analog

Sensor Interfacing & Signal Conditioning

## Waveform Control

PWM Drive & Waveform Generation

## Timing & Measurements

Signal Measurement with Timing & Counter Control

## Logic & Math

Customizable Logic & Math Functions

## Safety & Monitoring

Hardware Monitoring & Fault Detection

## Communications

Wired, Wireless & Encryption

## User Interface

Capacitive Touch Sensing & LCD Control

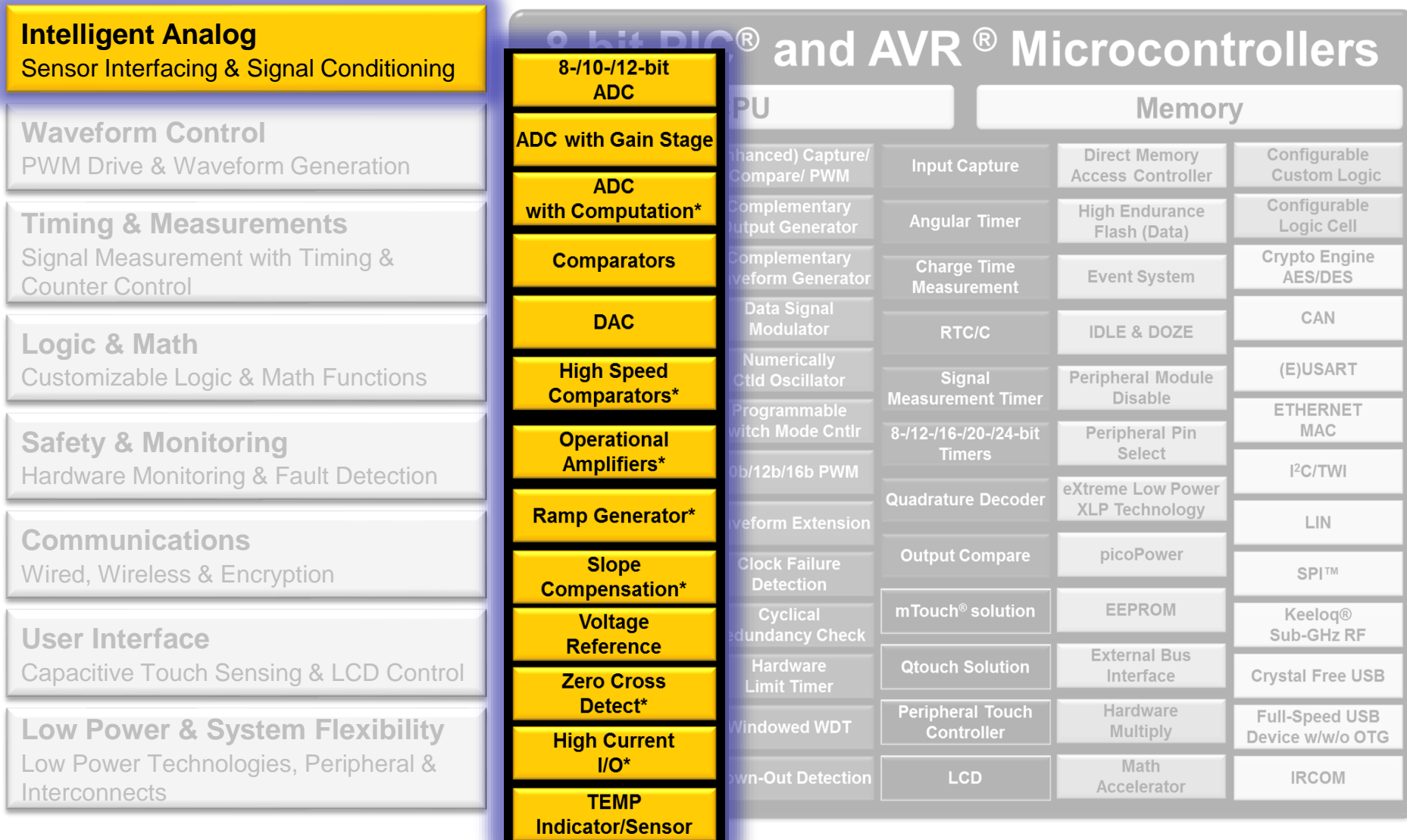
## Low Power & System Flexibility

Low Power Technologies, Peripheral & Interconnects

## 8-bit PIC® and AVR® Microcontrollers

CPU		Memory		
8-/10-/12-bit ADC	(Enhanced) Capture/Compare/ PWM	Input Capture	Direct Memory Access Controller	Configurable Custom Logic
ADC with Gain Stage	Complementary Output Generator	Angular Timer	High Endurance Flash (Data)	Configurable Logic Cell
ADC with Computation*	Complementary Waveform Generator	Charge Time Measurement	Event System	Crypto Engine AES/DES
Comparators	Data Signal Modulator	RTC/C	IDLE & DOZE	CAN
DAC	Numerically Ctd Oscillator	Signal Measurement Timer	Peripheral Module Disable	(E)USART
High Speed Comparators*	Programmable Switch Mode Cntrl	8-/12-/16-/20-/24-bit Timers	Peripheral Pin Select	ETHERNET MAC
Operational Amplifiers*	10b/12b/16b PWM	Quadrature Decoder	eXtreme Low Power XLP Technology	I <sup>2</sup> C/TWI
Ramp Generator*	Waveform Extension	Output Compare	picoPower	LIN
Slope Compensation*	Clock Failure Detection	mTouch® solution	EEPROM	SPI™
Voltage Reference	Cyclical Redundancy Check	Qtouch Solution	External Bus Interface	Keeloq® Sub-GHz RF
Zero Cross Detect*	Hardware Limit Timer	Peripheral Touch Controller	Hardware Multiply	Crystal Free USB
High Current I/O*	Windowed WDT	LCD	Math Accelerator	Full-Speed USB Device w/w/o OTG
TEMP Indicator/Sensor	Brown-Out Detection			IRCOM

# AVR<sup>®</sup> MCU 也內建豐富的類比周邊





# AVR<sup>®</sup> DA Product Family 的關鍵特點

- **AVR<sup>®</sup> CPU**
- **24 MHz @ 1.8V to 5.5V – 相當於 24 MIPS**
- **Memory**
  - 32KB – 128KB Flash
  - 4KB – 16KB SRAM
  - 512B EEPROM
- **Pin and Packages**
  - 28 to 64 pins
  - SPDIP, SSOP, SOIC, TQFP, VQFN
- **Temperature Grade**
  - Standard: -40°C to 85°C
  - Extended: -40°C to 125°C
- **Key Features**
  - 12-bit differential ADC
  - 10-bit DAC
  - Analog Comparators w/individual DACREF
  - Zero Cross Detectors
  - Configurable Custom Logic (CCL)
  - Event System
  - USART, SPI, TWI
  - 16- and 12-bit Timer/Counters
  - Peripheral Touch Controller (PTC)

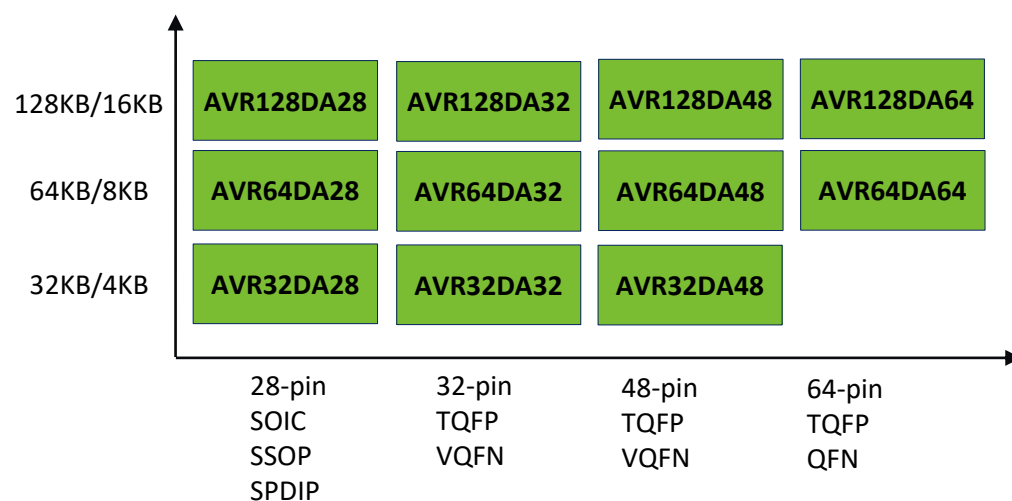


# AVR DA Microcontrollers 族系列表



## Messaging

Our first Next-Generation AVR MCU with high pin count and high memory to extend the existing portfolio. Offering integrated analog CIPs with high Flash-to-SRAM ratio, which enables Real-Time Control, Connectivity, HMI and Safety Critical Applications.



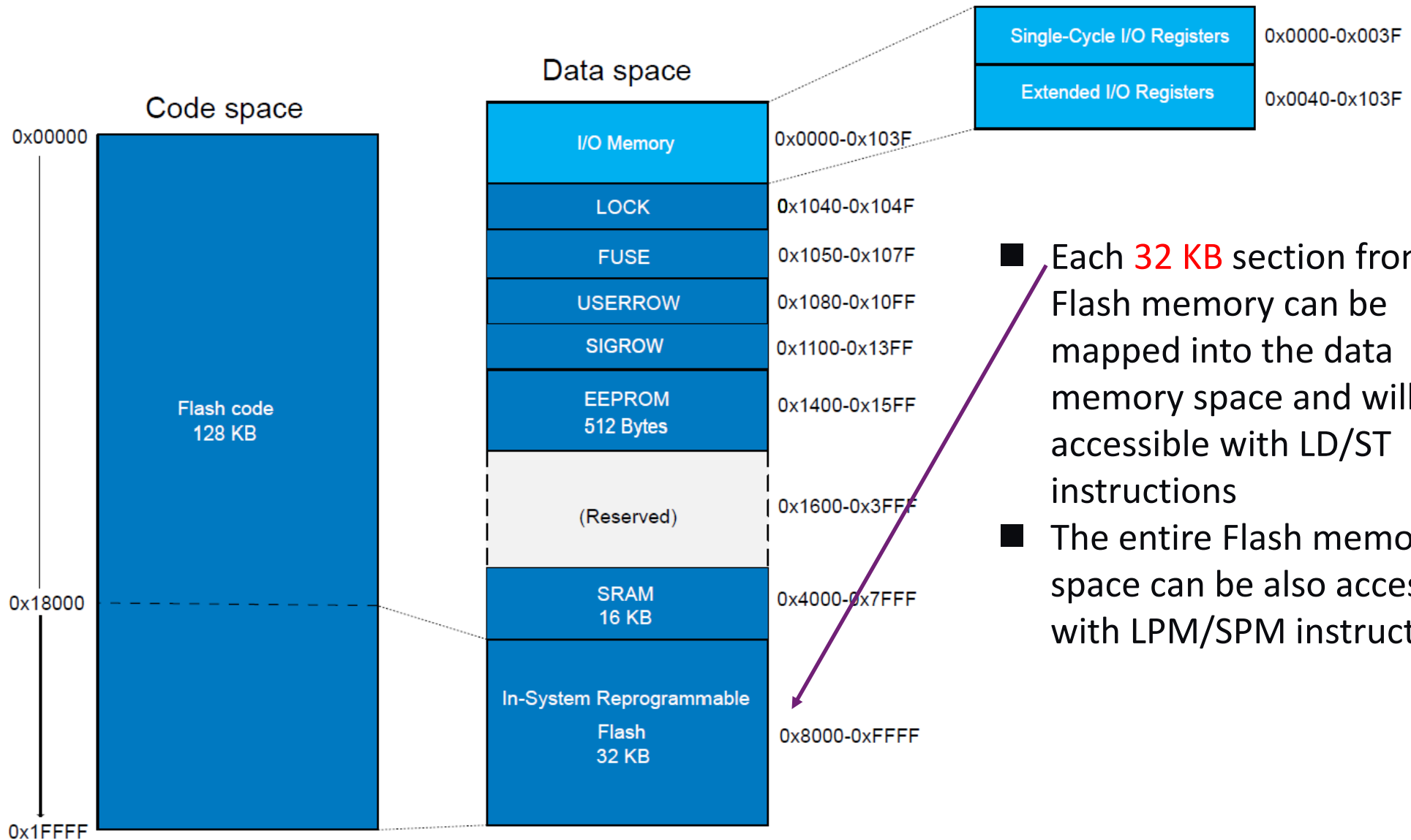
## Features

Functional-Safety Ready at launch!  
22ch 12bit Differential ADC (130Ksps)  
10-bit DAC  
3x Analog Comparators  
3x Zero Cross Detectors  
Configurable Custom Logic (CCL)  
Peripheral Touch Controller (PTC)  
10ch Event System  
Cyclic Redundancy Check (CRC) Scan  
6x EUSART + 2xSPI + 2xI2C

# AVR128DA 系列 Flash Memory 的實體結構

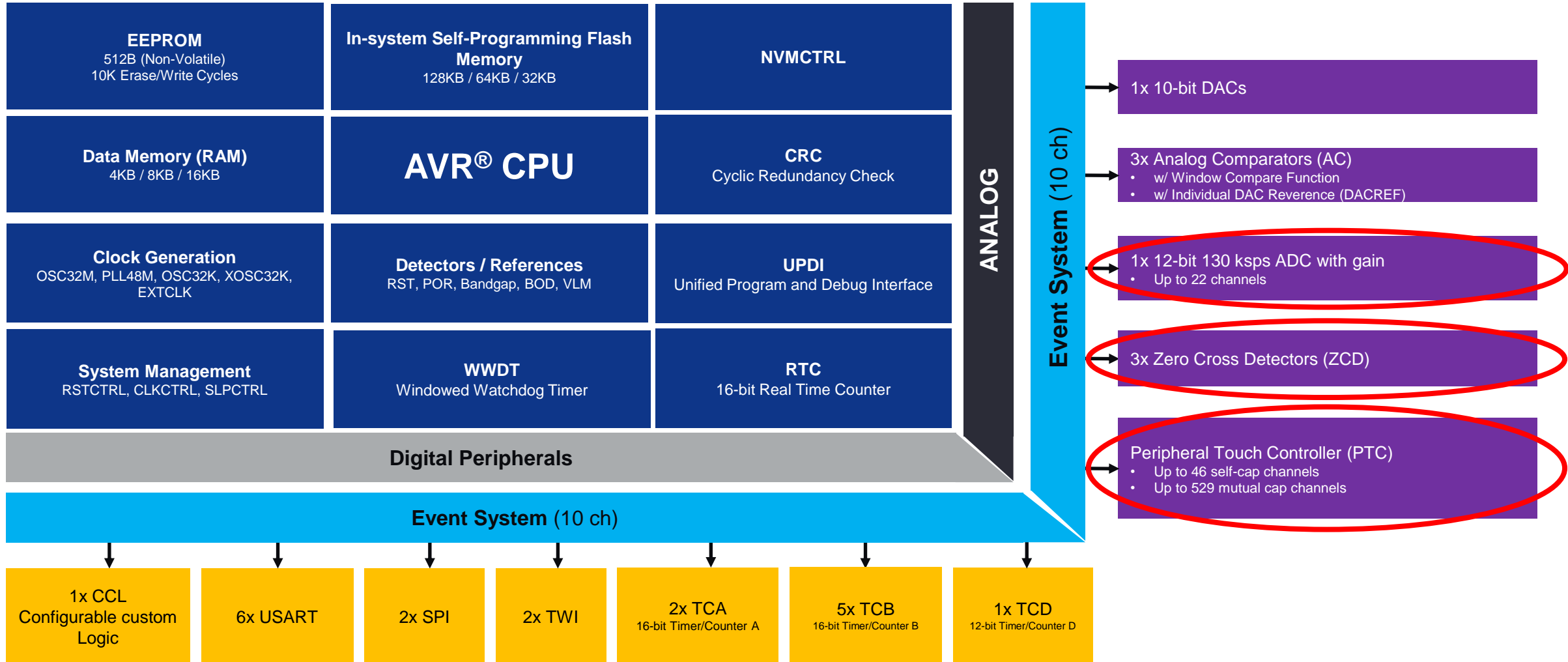
Property	AVR128DA28 AVR128DA32 AVR128DA48 AVR128DA64
Size	128 KB
Page size	512B
Number of pages	256
Start address in data space	0x8000
Start address in code space	0x0

# AVR® DA MCU Memory Map



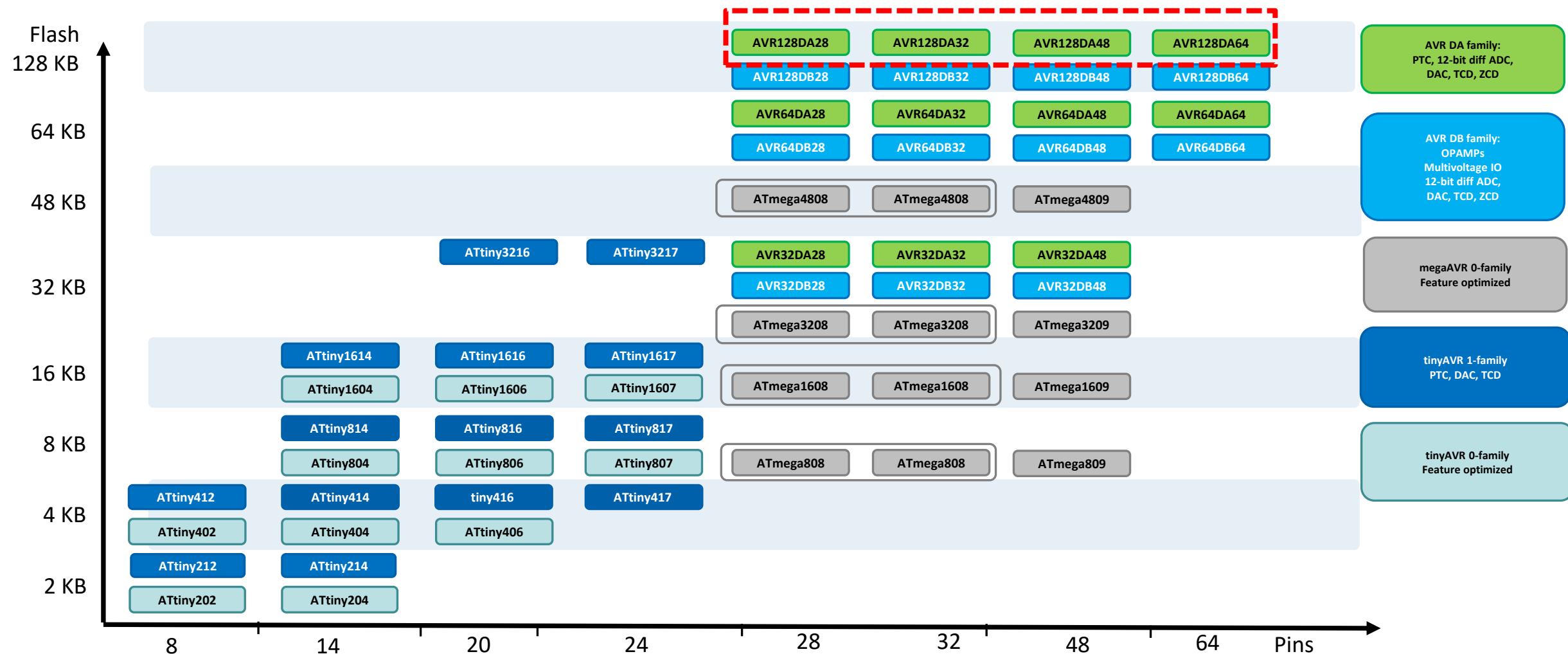
- Each 32 KB section from Flash memory can be mapped into the data memory space and will be accessible with LD/ST instructions
- The entire Flash memory space can be also accessed with LPM/SPM instruction

# AVR® DA Family 方塊圖



The block diagram is showing the maximum numbers of peripherals and memory.

# AVR® MCU Family Portfolio Overview



# AVR® DA Part Number 的編碼範例

AVR[Memory][Family][Pincount][media]-[Temp]/[Package style]

## • Example 1:

- AVR128DA64T-I/PT
- AVR = Architecture
- 128 = Flash Memory
- DA = Family
- 64 = Pins
- T = Tape&Reel
- I = Ind Temp Grade, -40 to 85C
- PT = TQFP

## • Example 2

- AVR128DA64-E/PT
- AVR = Architecture
- 128 = Flash Memory
- DA = Family
- 64 = Pins
- No letter is bulk media – either Tube or Tray
- E = Ext Temp Grade, -40 to 125C
- PT = TQFP

\*Automotive part numbers will have a VAO suffix after the package style, e.g.

AVR128DA64T-I/PTVAO

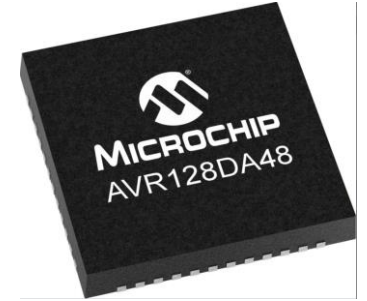
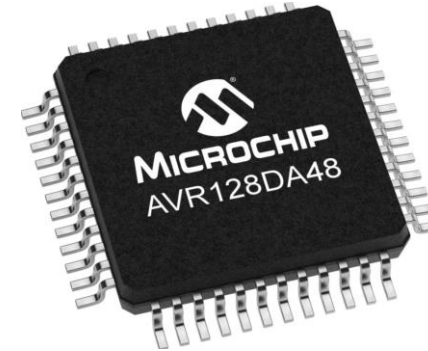


# 不同型號的 AVR<sup>®</sup> DA MCU 內建周邊比較

Device	AVR128/64/32DA28	AVR128/64/32DA32	AVR128/64/32DA48	AVR128/64DA64
Pins	28	32	48	64
GPIO	23	27	41	56
External Interrupts	23	27	41	56
Event System ch.	8	8	10	10
CCL LUTs	4	4	6	8
RTC	1	1	1	1
16-bit Timer/Counter type A (TCA)	1	1	2	2
16-bit Timer/Counter type B (TCB)	3	3	4	5
12-bit Timer/Counter type D (TCD)	1	1	1	1
PWM channels	8	8	12	13
USART	3	3	5	6
SPI	2	2	2	2
TWI / I <sup>2</sup> C	1	2	2	2
12-bit diff ADC (input pins)	1 (10)	1 (14)	1 (18)	1 (22)
10-bit DAC (output buffer)	1	1	1	1
Analog Comparators with DACREF	3	3	3	3
Zero Cross Detectors (ZCD)	1	1	2	3
Peripheral Touch Controller (PTC) (self/mutual cap ch)	Yes (18/81)	Yes (22/121)	Yes (32/256)	Yes (46/529)

# 新推出的 8-bit AVR® and PIC® MCU 涵蓋的封裝

- **Package: 8 / 14 / 20 / 28 / 48 Pin be used by PIC & AVR - all package sizes in mm**
  - **8-Pin:** SOIC 6x4.9 and DFN 3x3x0.9
  - **14-Pin:** TSSOP 5x 6.4 and SOIC 6x8.65
  - **20-Pin:** SSOP 7.8x5.3, SOIC 10.3 x 12.8, VQFN 4x4 VQFN and VQFN 3x3x0.9
  - **28-pin:** SPDIP 7.8x34.7, SOIC 10.3x17.5 and SSOP 10.2 x 7.8 mm, VQFN 4x4mm
  - **48-pin:** TQFP 7x7x1 and VQFN 6x6x0.9
- **Extra PIC packages – 16 / 40 / 44Pin**
  - **16-Pin:** VQFN 3x3x0.9,
  - **20-Pin:** PDIP 7.8x26.2,
  - **28-Pin:** VQFN 6x6
  - **40-Pin:** PDIP 15.8x53.2 and VQFN 5x5,
  - **44-Pin:** TQFP 10x10x1,
- **Extra AVR packages – 24 / 32 / 64Pin**
  - **24-pin:** VQFN 4x4 mm (only available in tinyAVR 0/1/2, not planned for future families)
  - **32-Pin:** TQFP 7x7x1.0mm and VQFN 5x5x0.9mm,
  - **64-Pin:** TQFP 10x10x1mm64 and QFN 9x9x0.9mm,



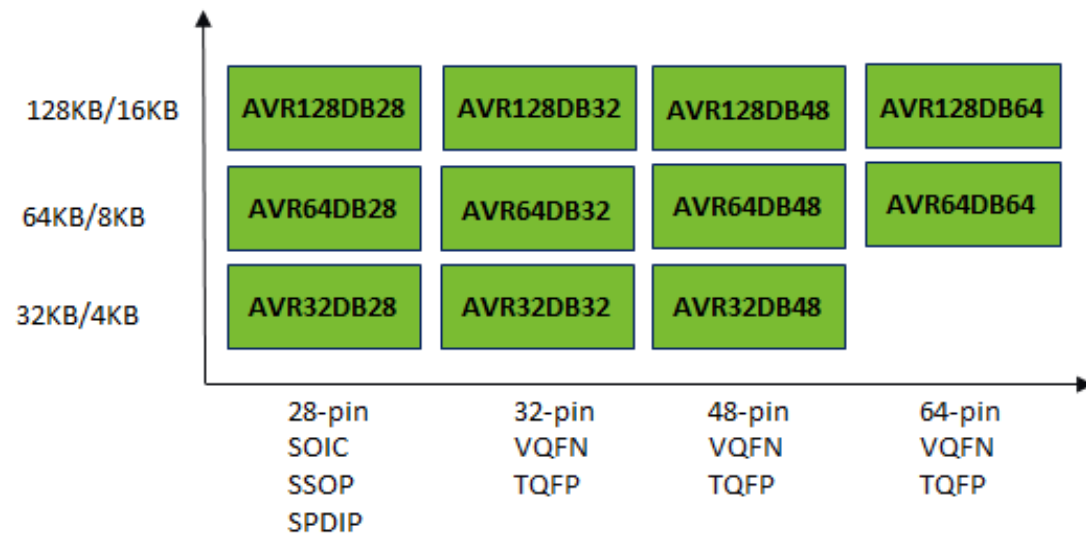
# 其他新推出的AVR® MCU 系列 (自行閱讀)

---

# AVR DB

## Messaging

Perfect for edge nodes and as companion device in complex designs. Feature rich entry level AVR for new designs. Brings **Multi Voltage I/O** and memory scaled for complex interconnected applications or operating systems.



## Features

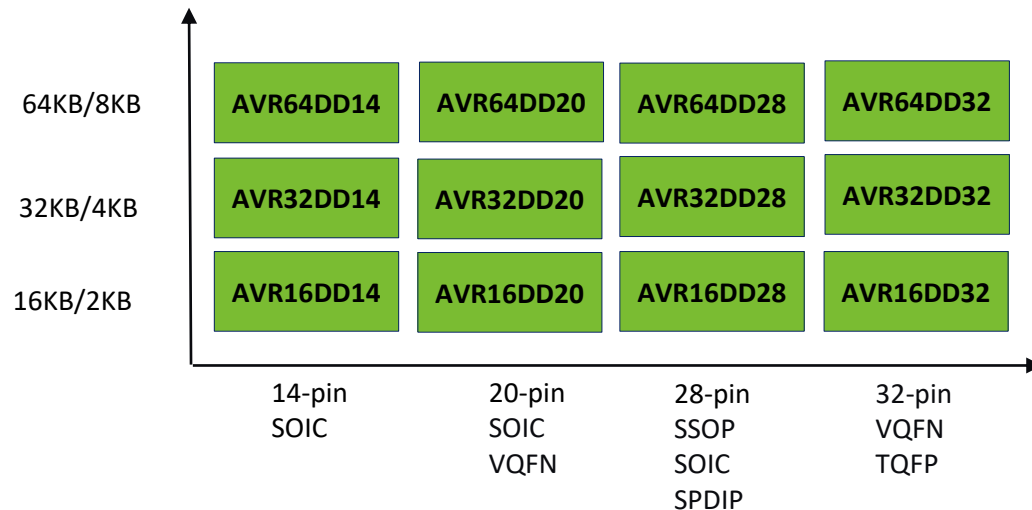
- **Multi-Voltage I/O on one port**
- **3x OPAMPs**
- **22ch 12bit Differential ADC (130Ksps)**
- **3x Analog Comparators**
- **3x Zero Cross Detectors**
- **10-bit DAC**
- **6x EUSART + 2xSPI + 2xI2C**
- **10ch Event System**
- **Cyclic Redundancy Check (CRC) Scan**
- **Ext. Crystal with failure detect**

# AVR DD Microcontrollers

## Messaging

Memory and feature rich entry level AVR for new designs. Includes the integrated level shifter peripheral (MVIO) for easy multi-voltage domain operation. Perfect device for edge nodes and as a companion device in more complex designs.

Pin compatible with AVR DA and AVR DB.



## Features

Multi-Voltage I/O on one port

27ch 12bit Differential ADC (130Ksps)

1x Analog Comparators

1x Zero Cross Detectors

10-bit DAC

1xTCA + 3xTCB + 1xTCD

2x EUSART + 1xSPI + 1xI2C

6 channel Event System

4 CCL LUTs

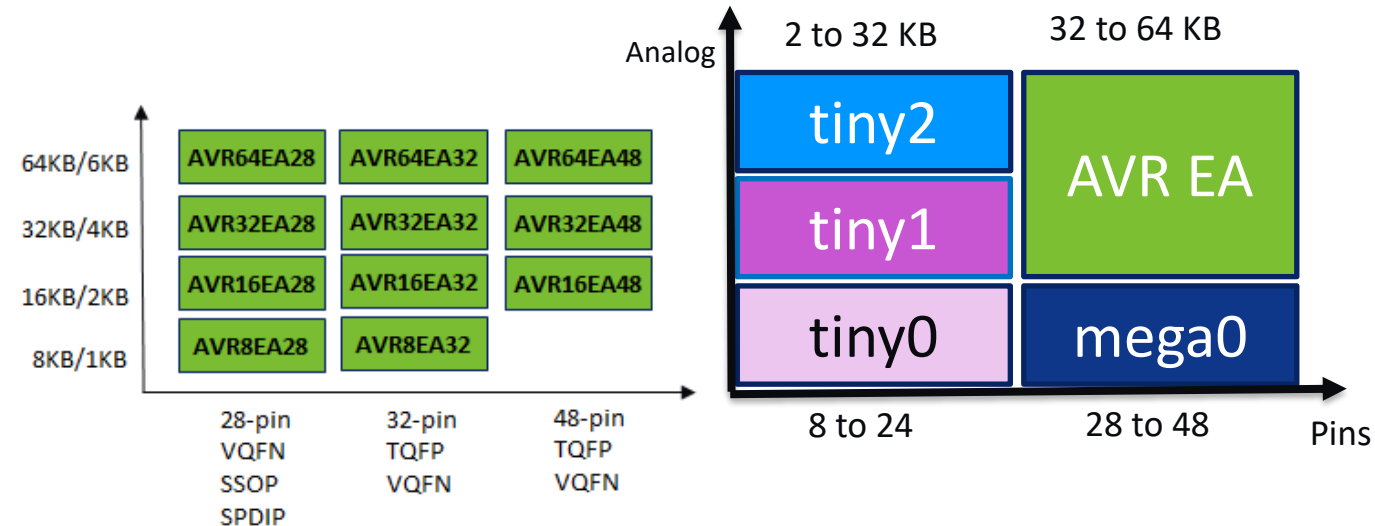
Cyclic Redundancy Check (CRC) Scan

Ext. Crystal Osc with failure detect

# AVR EA Microcontrollers

## Messaging

Cost efficient upgrade path from the megaAVR 0-family, offering more analog to support sensor node applications in noisy and harsh environments and secondary safety monitoring in larger systems.



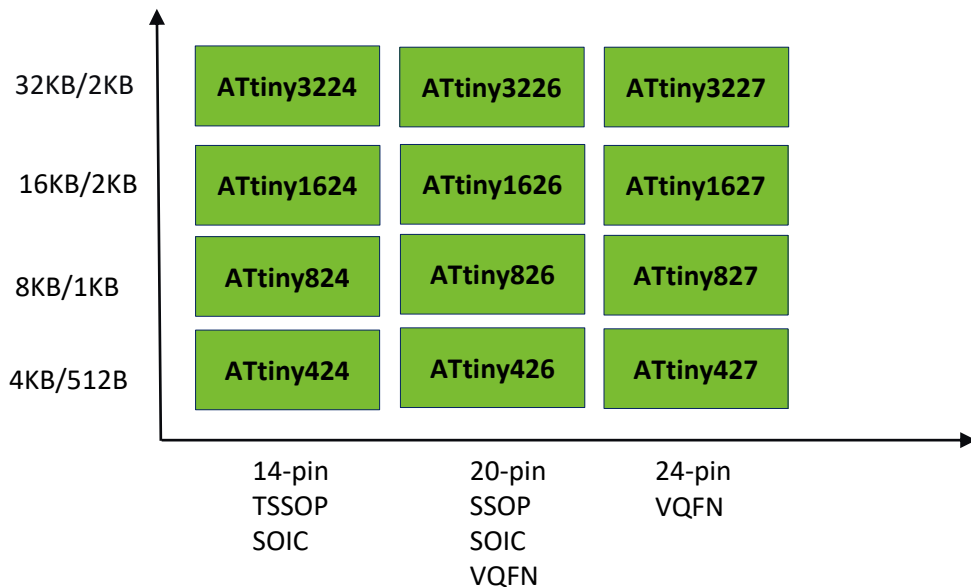
## Features

- 12-bit differential ADC with Computation
- Programmable Gain Amplifier, up to 16x
- 3x USART +I2C +SPI
- Configurable Custom Logic (CCL)
- 2x Analog Comparator w/internal DACREF
- 1x 10-bit DAC (output)
- 8 channel Event System
- Cyclic Redundancy Check (CRC) Scan
- 6x 16-bit Timer/Counters
- High speed external crystal oscillator (XOSCHF)
- Clock Failure Detection (CFD)
- Flash memory with dedicated RWW section.

# tinyAVR 2 Microcontrollers

## Messaging

Designed to meet the requirements of small signal sensor node applications, sensors in noisy and harsh environments and secondary safety monitoring by introducing a highly flexible 12-bit differential ADC with computation and Programmable Gain Amplifier (PGA).



## Features

- 12-bit differential ADC with Computation (375 Ksps)
- Programmable Gain Amplifier, up to 16x
- 2x USART +I2C +SPI
- Configurable Custom Logic (CCL)
- 1x Analog Comparator w/ internal DACREF
- 6 channel Event System
- Cyclic Redundancy Check (CRC) Scan
- 3x 16-bit Timer/Counters
- Functional Safety Ready



# AVR128DA MCU可用的開發工具及環境

---

# Simplified Software Development

PIC® MCU and AVR® MCU



*Support for All PICs & All New AVRs*

**Community**

**Code  
Generation**

**Development  
Environment**



*No plans for EOL. Support for All AVRs.*

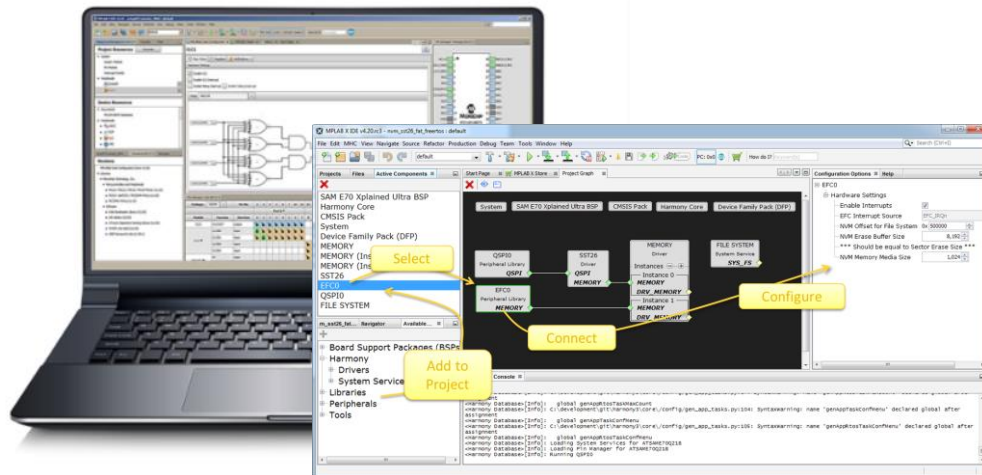
# Easy & Fast GUI Configuration Tools

## MCC & START

- **Configure MCU peripherals in minutes!**
  - MCC supports 8-/16-bit PIC and **new 8-bit AVR MCUs**
  - START supports 8-bit AVR MCUs
- **Generates optimized C code**
- **Minimizes reliance on datasheet**
- **FREE. Try them today by clicking the links below.**



[www.microchip.com/mcc](http://www.microchip.com/mcc)



[start.atmel.com](http://start.atmel.com)



# Microchip MCU 開發的 3 個必備軟體工具



## MPLAB X INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)

MPLAB® X Integrated Development Environment (IDE) is an expandable, highly configurable software program that incorporates powerful tools to help you discover, configure, develop, debug and qualify...

[Learn More](#)



Available as free, unrestricted-use downloads, our award-winning MPLAB® XC C Compilers are comprehensive solutions for your project's software development.

[Learn More](#)









## MPLAB CODE CONFIGURATOR

MPLAB Code Configurator supports 8-bit, 16-bit and 32-bit PIC® microcontrollers. MCC is incorporated into both the downloadable MPLAB X Integrated Development Environment (IDE) and the cloud-based MPLAB...

[Learn More](#)

# Microchip MCU 開發階段建議的除錯/燒錄器

 <p>MPLAB® PICKIT™ 5 In-Circuit Debugger</p> <p><a href="#">Learn More</a></p> <p><a href="#">Add to Cart</a></p>	 <p>MPLAB® ICD 5 In-Circuit Debugger/Programmer</p> <p><a href="#">Learn More</a></p> <p><a href="#">Add to Cart</a></p>	 <p>MPLAB® ICE 4 In-Circuit Emulator, Programmer and Debugger</p> <p><a href="#">Learn More</a></p> <p><a href="#">Add to Cart</a></p>	 <p>MPLAB® ICD 4 In-Circuit Debugger</p> <p>The MPLAB® ICD 4 In-Circuit Debugger/Programmer is Microchip's fastest, cost-effective debugging and programming tool for PIC® Microcontrollers (MCUs), Microprocessors (MPUs) and dsPIC® Digital Signal</p> <p><a href="#">Learn More</a></p>	 <p>MPLAB® SNAP</p> <p>The MPLAB® Snap In-Circuit Debugger/Programmer allows affordable, fast and easy debugging and programming of PIC®, dsPIC® and AVR flash MCUs and MPUs, using the powerful graphical user</p> <p><a href="#">Learn More</a></p> <p><a href="#">Add to Cart</a></p>	 <p>MPLAB® PICKIT™ 4 In-Circuit Debugger</p> <p>The MPLAB® PICKIT™ 4 In-Circuit Debugger/Programmer allows fast and easy debugging and programming of PIC®, dsPIC®, AVR, SAM and CEC flash microcontrollers and microprocessors.</p> <p><a href="#">Learn More</a></p> <p><a href="#">Add to Cart</a></p>
--	---	--	---	---	--

# Curiosity Development Ecosystem

## Fully Integrated Platform for 8-, 16- and 32-bit PIC & SAM MCUs

### Curiosity

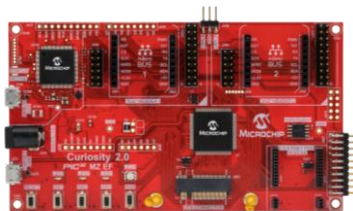
- Standardized features & interfaces
- Expansion Headers for MikroE Clicks



Curiosity High Pin Count (HPC)  
Development Board  
DM164136



dsPIC33CH Curiosity  
Development Board  
DM330028-2



PIC32MZ EF 2.0 Curiosity  
Development Board  
DM320209

### Curiosity *Nano*

- Lower cost prototyping board
- Standardized pinout
- Full programming and debugging support



SAM D21 Curiosity Nano  
DM320119



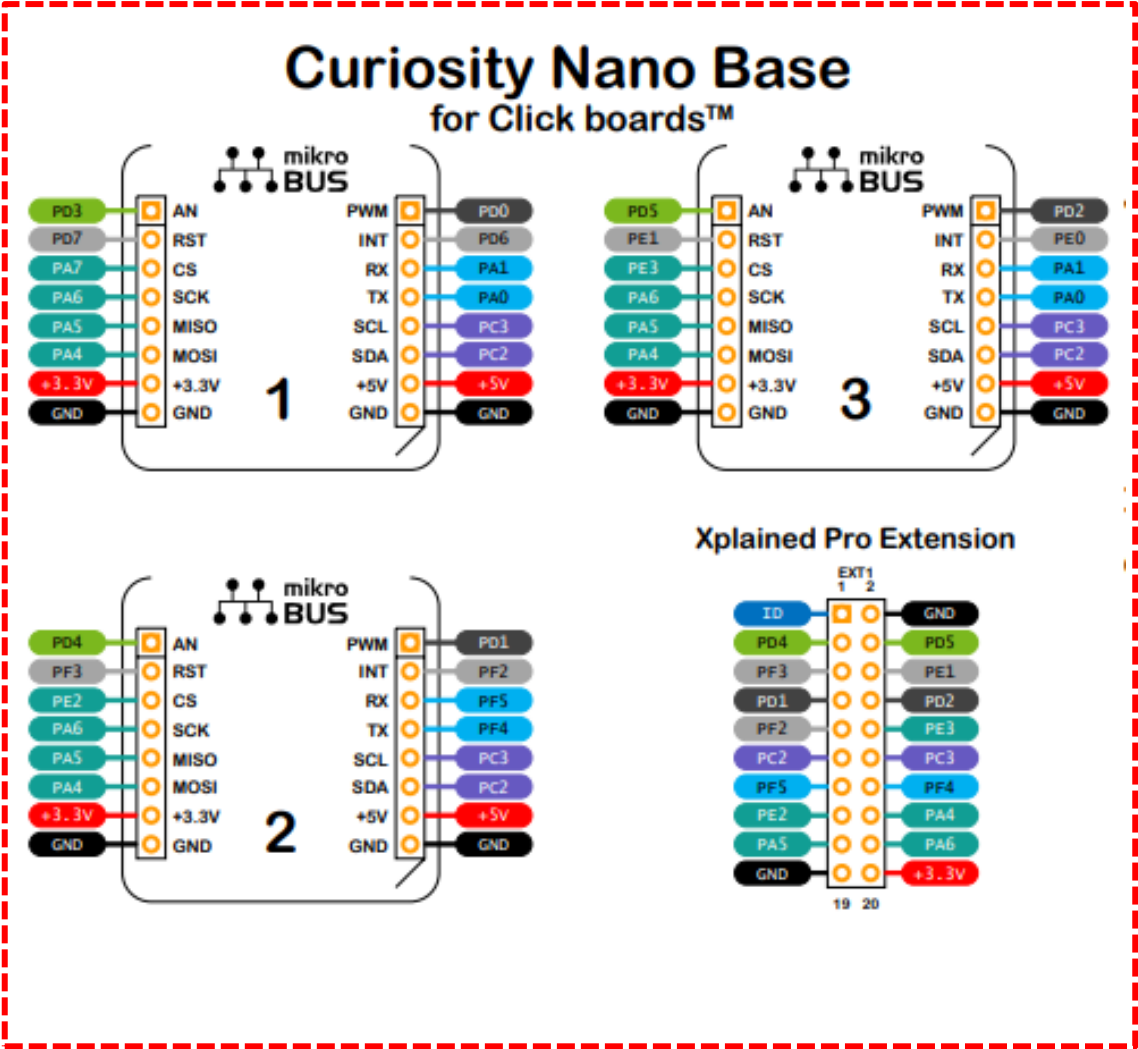
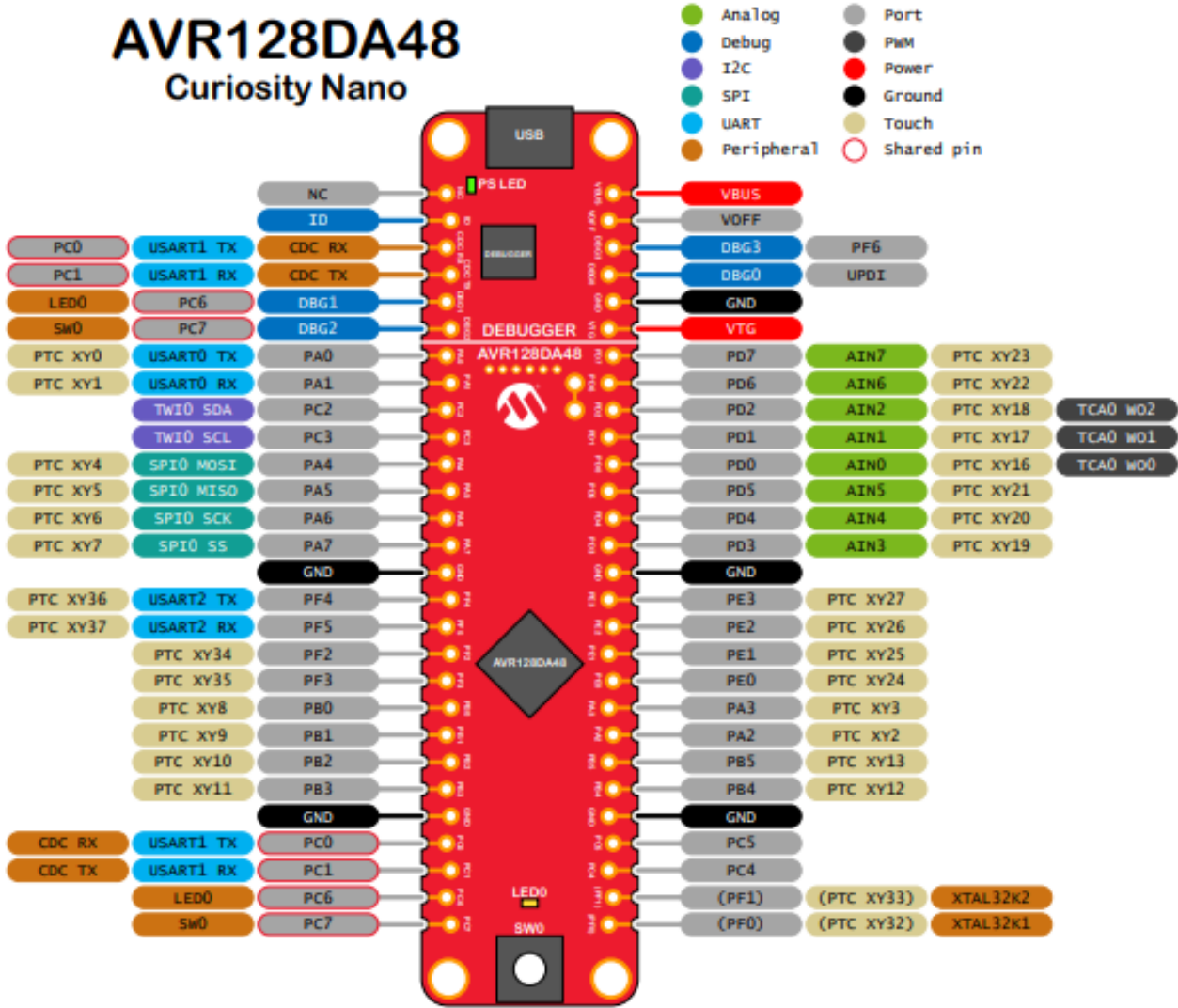
AVR128DA48 CURIOSITY NANO  
DM164151



ATmega4809 Curiosity Nano  
DM320115



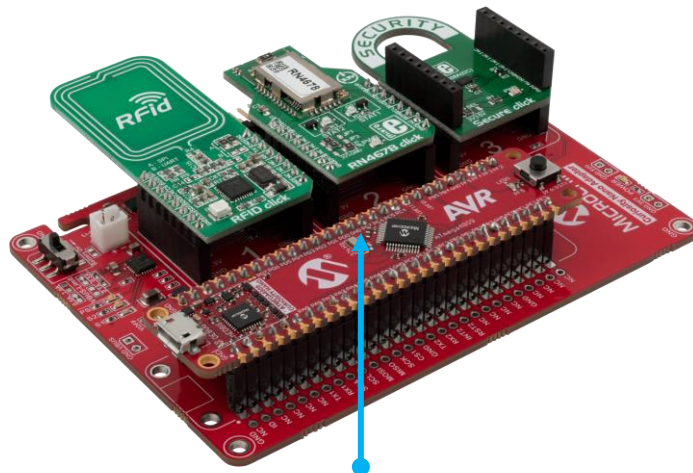
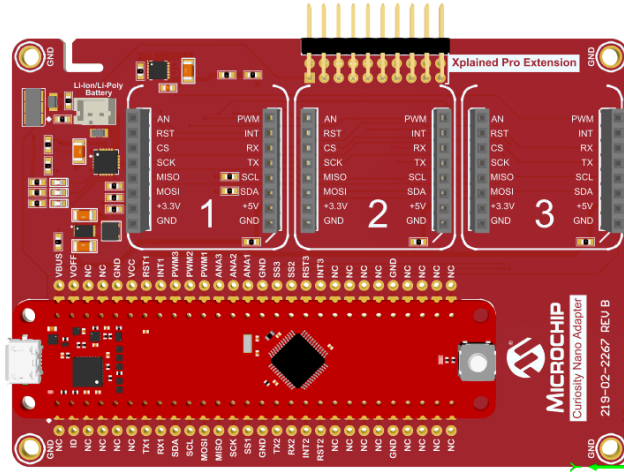
# Curiosity Nano : 預先設定的統一接腳配置





# Curiosity Nano Base Board for Click™ Boards

Part No: AC164162 (\$35.41 USD – Sep/2023)



Curiosity Nano Board Footprint

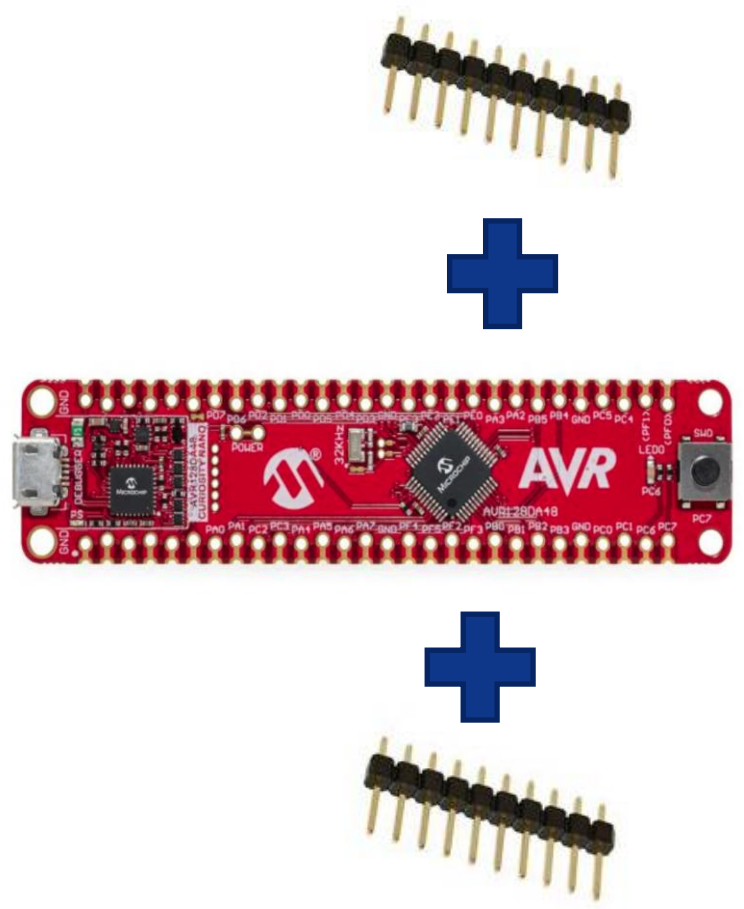
\*Xplained Nano Not Supported

- **Key Features**

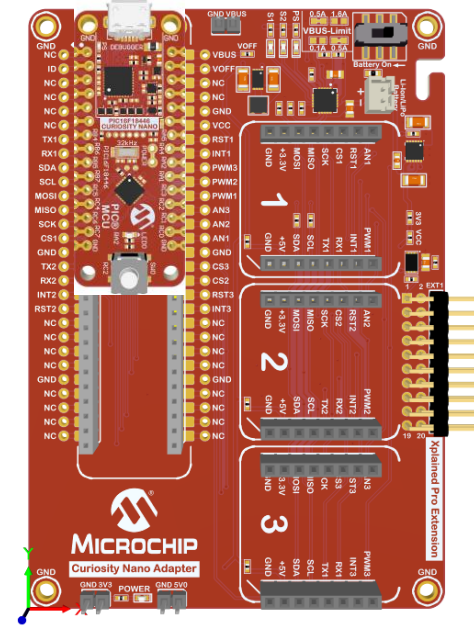
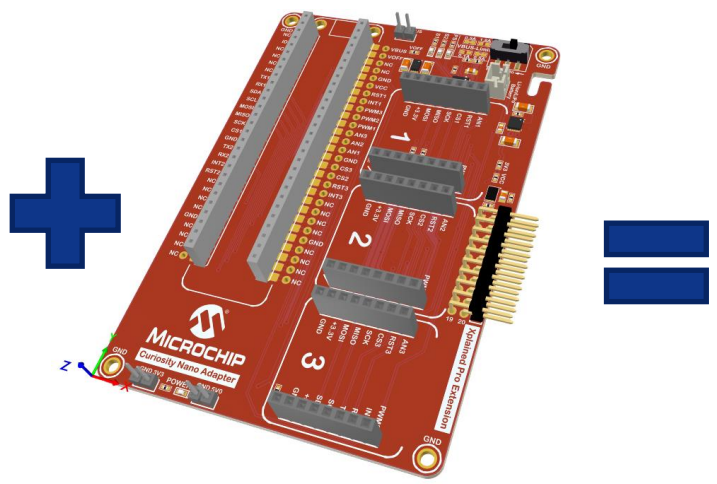
- Single USB connection
- Access to all I/O pins
- Allow the user to connect to
  - Clicks™ and Xplained PRO Extensions
  - Support up to 3 Clicks™ modules
- Flexible Power Options
  - MCHP Buck regulator @ 3.3V
  - MCHP Boost regulator @ 5V
  - Battery power option

# Curiosity Nano Mounting Headers

Insert headers into staggered footprints



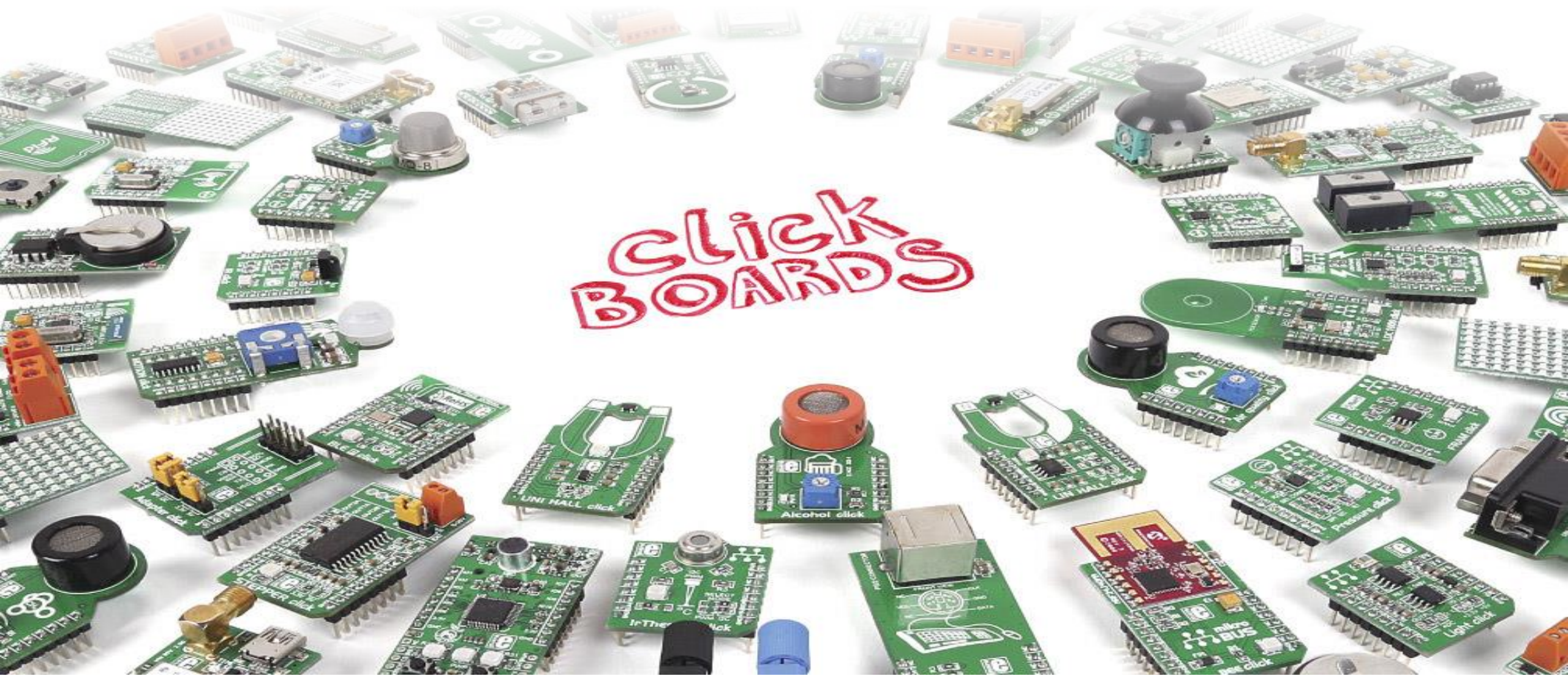
Mount Curiosity Nano on Base board





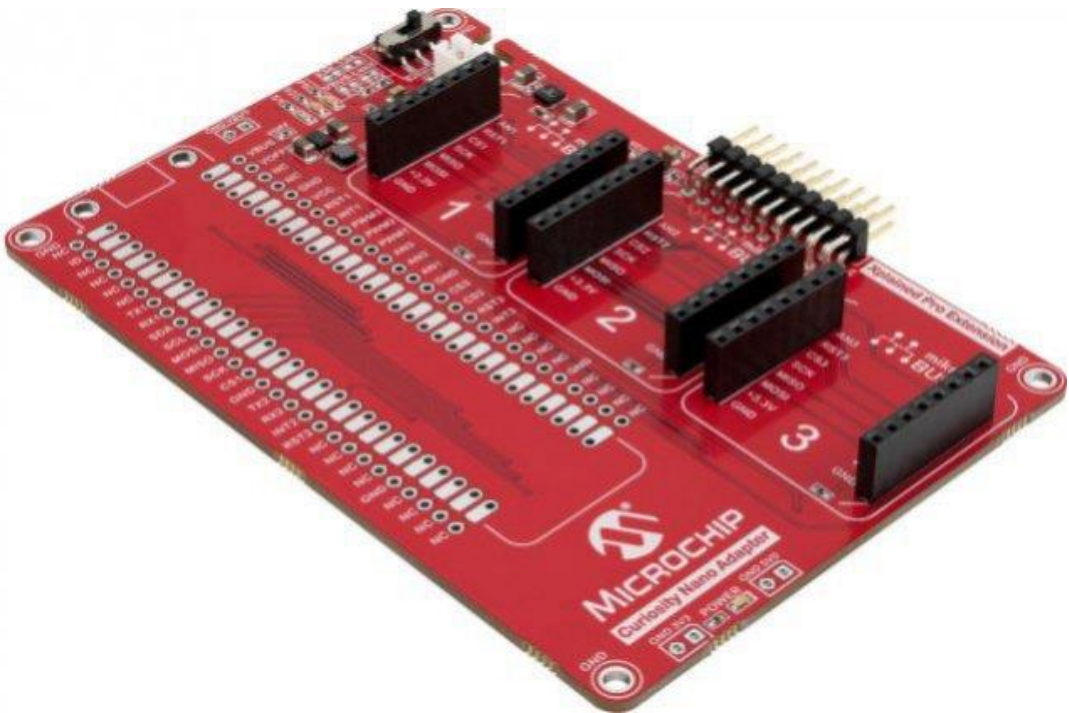
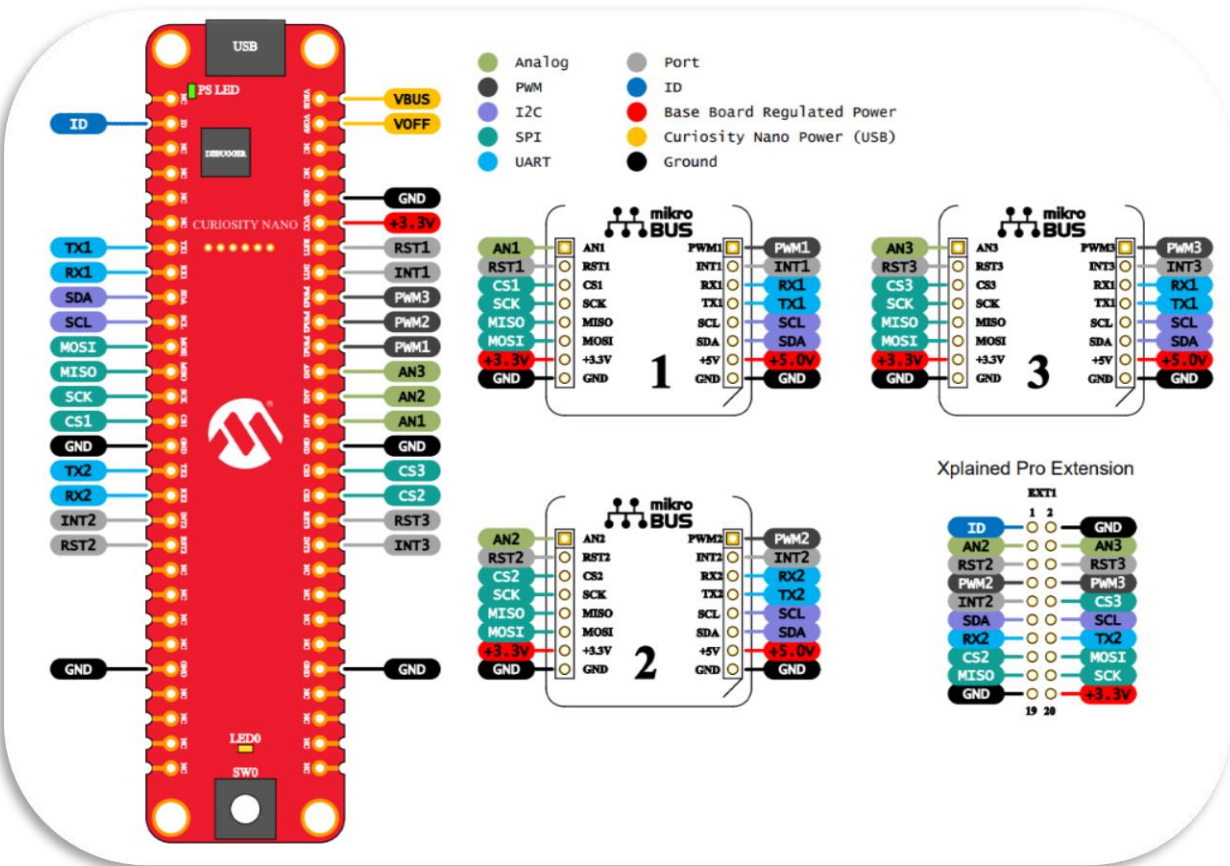
# MIKROE Click boards™

[www.mikroe.com/click](http://www.mikroe.com/click)





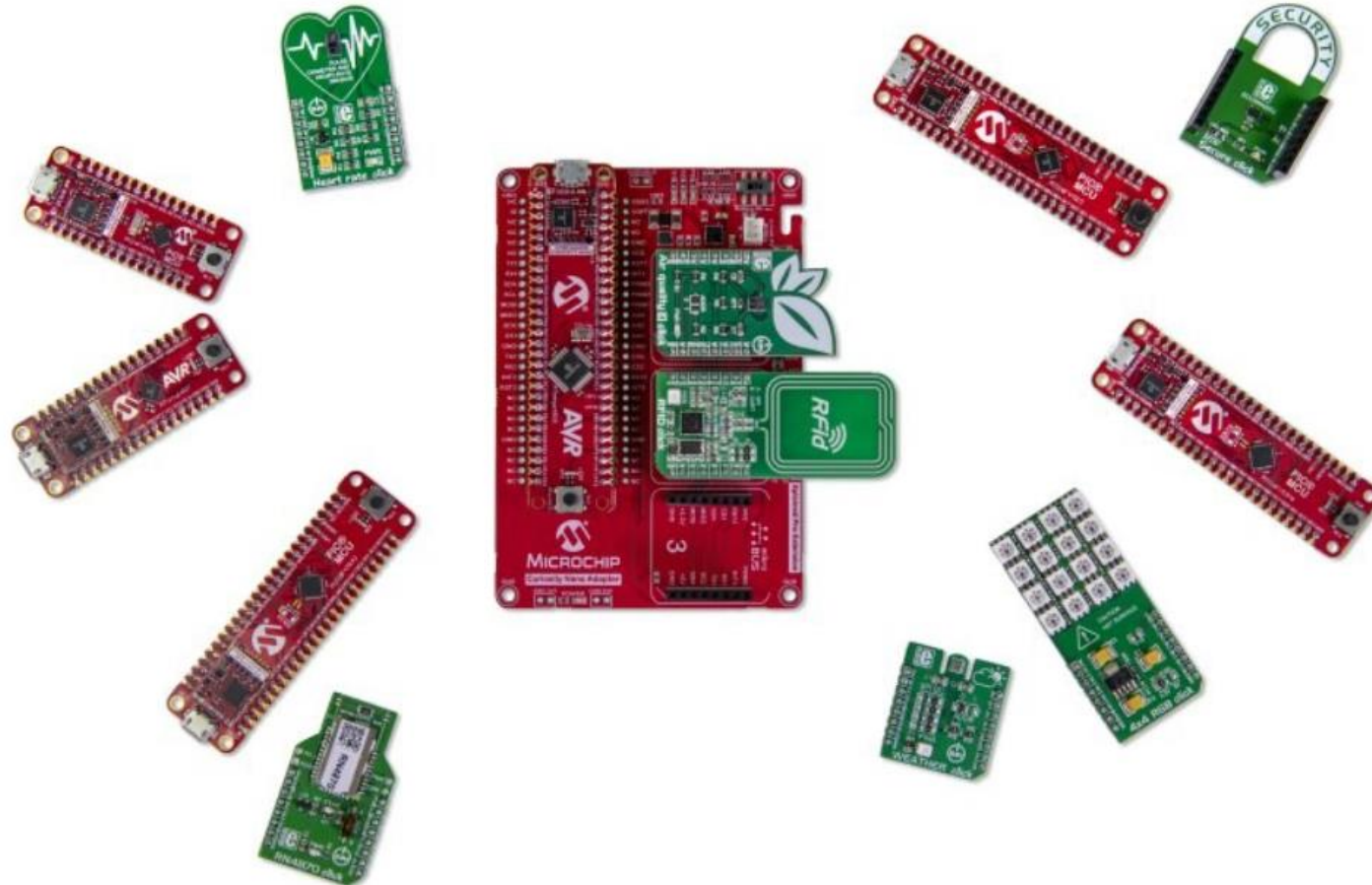
# Curiosity Nano Base is the Great Reference .. But



# Curiosity Nano Base has very limited function

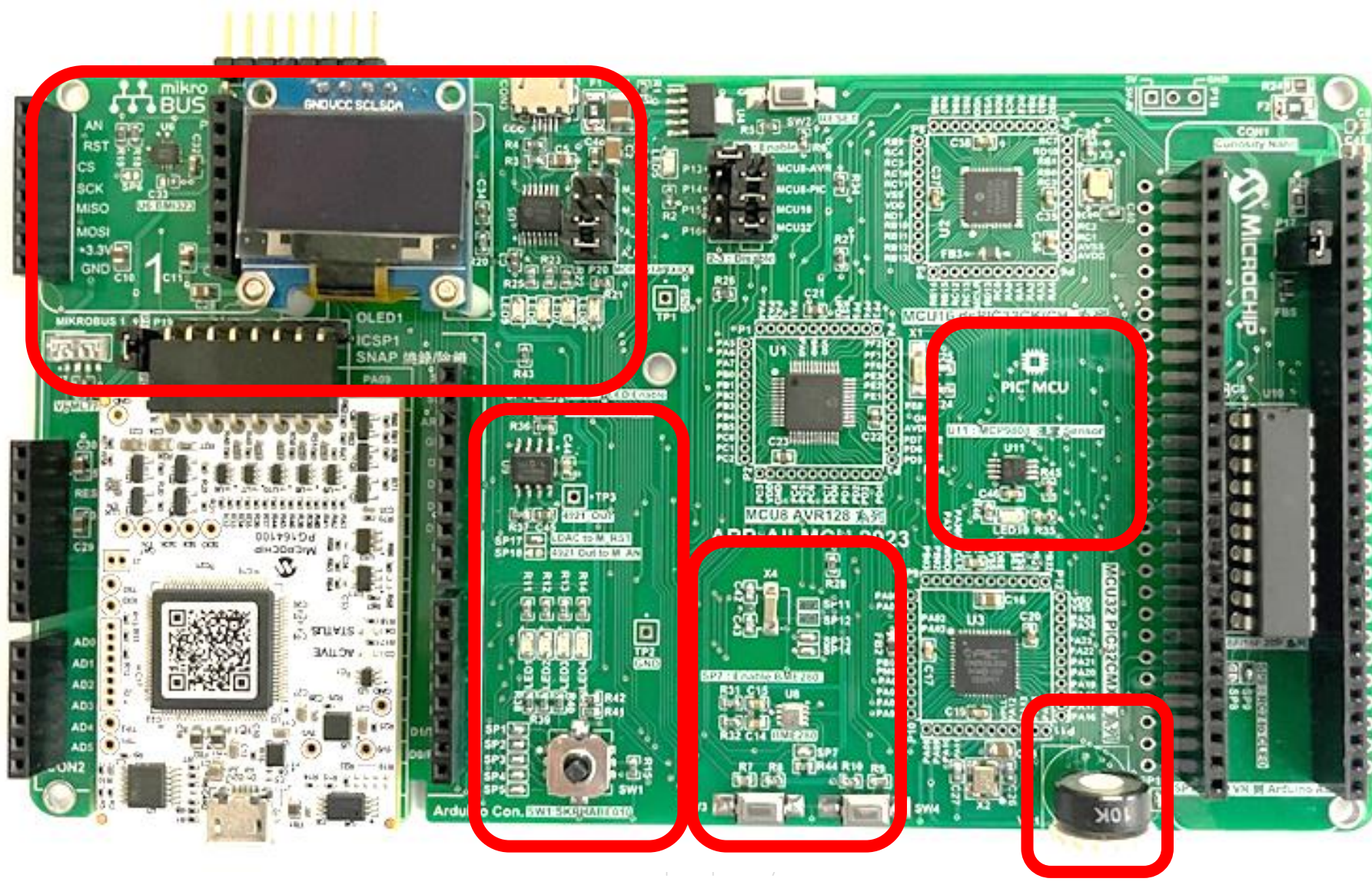
Need both Curiosity Nano and Click Board together to get a functional System

Even .... It's just for a simple experiment





改進的方式：先把一些常用或必要的周邊內建在實驗版上  
不夠的部分再用 Click™ Boards 或是 Arduino 擴充板來擴增



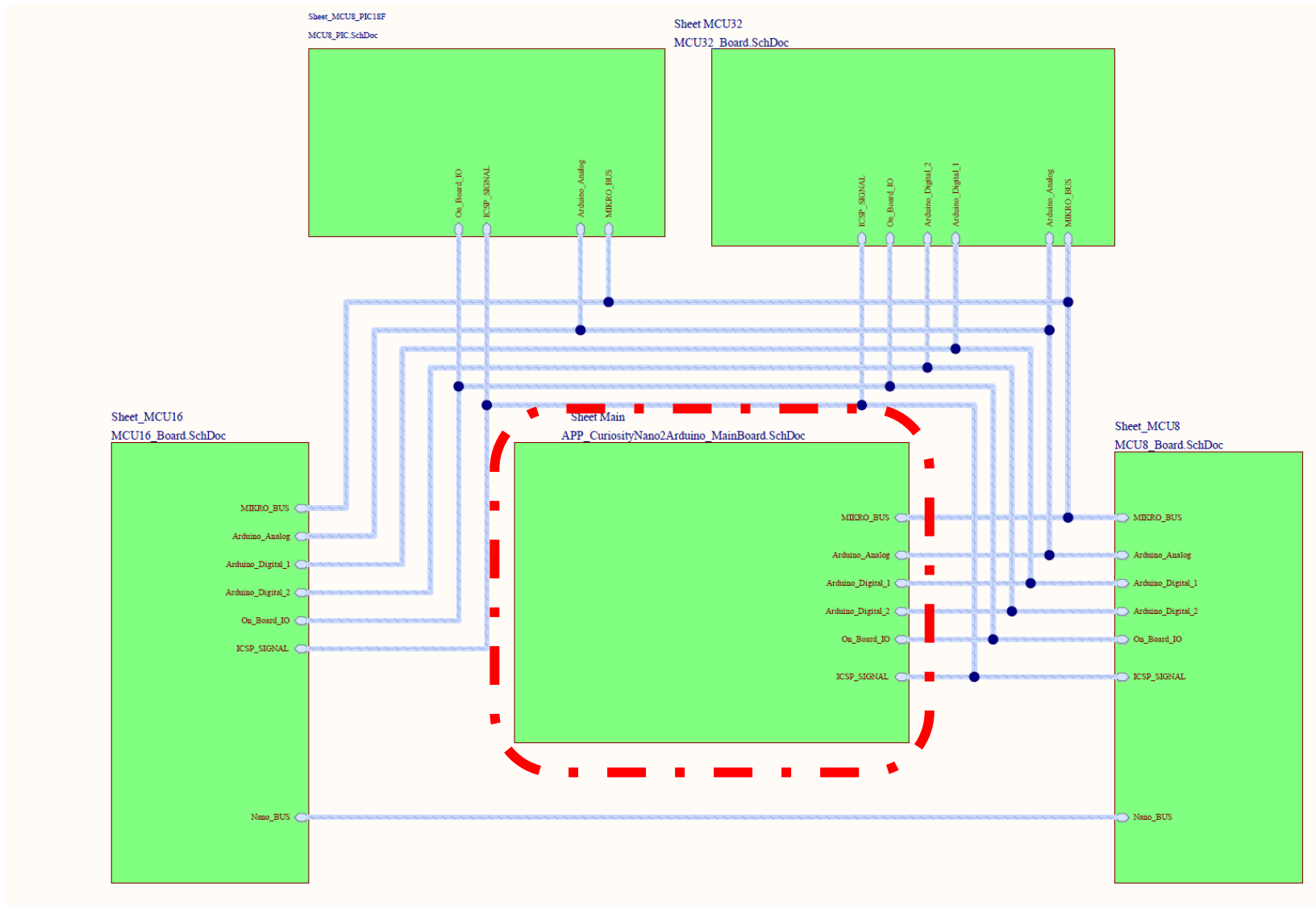
# APP-AI MCU 2023 各種 MCU 的共用周邊 (on-board Peripherals )

- ✓ 一個 I2C 介面的六軸 IMU - BOSCH BMI323
- ✓ 一個 I2C 介面的 Lighting Sensor – Vishay 的 VEML7700-TT
- ✓ 一個 I2C 介面的 Humidity sensor - BOSCH BME280
- ✓ 一個 I2C 介面的 溫度 Sensor – Microchip MCP9808
- ✓ 一個 I2C 介面的 OLED Display - 單色 128 \* 64
- ✓ 一個 SPI 介面的 DAC – Microchip MCP4921
- ✓ 兩個 WS2812B One-Wire Color LED
- ✓ 一個 MCP2221A 作實驗板上的 UART 以及 I2C 介面轉換至 USB 的介面 IC
- ✓ 一個 ALPS 的 SKRHABE010 五向開關
- ✓ LED \* 4
- ✓ Switch Button \* 2

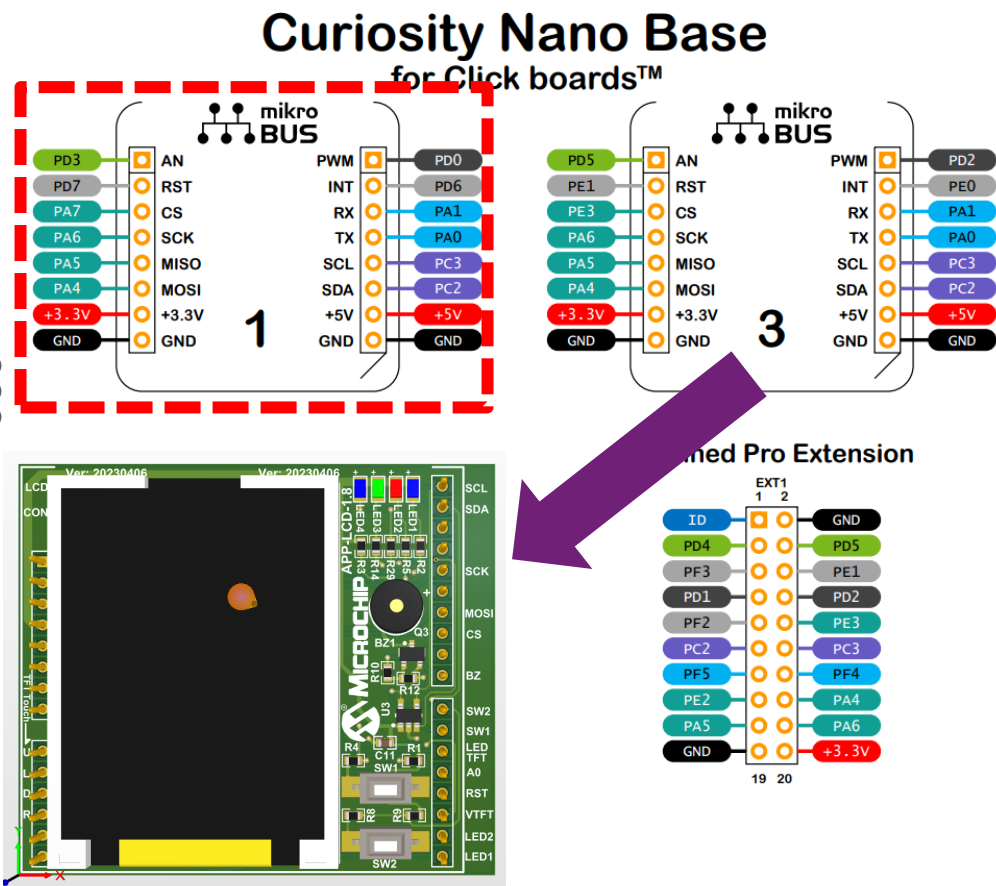
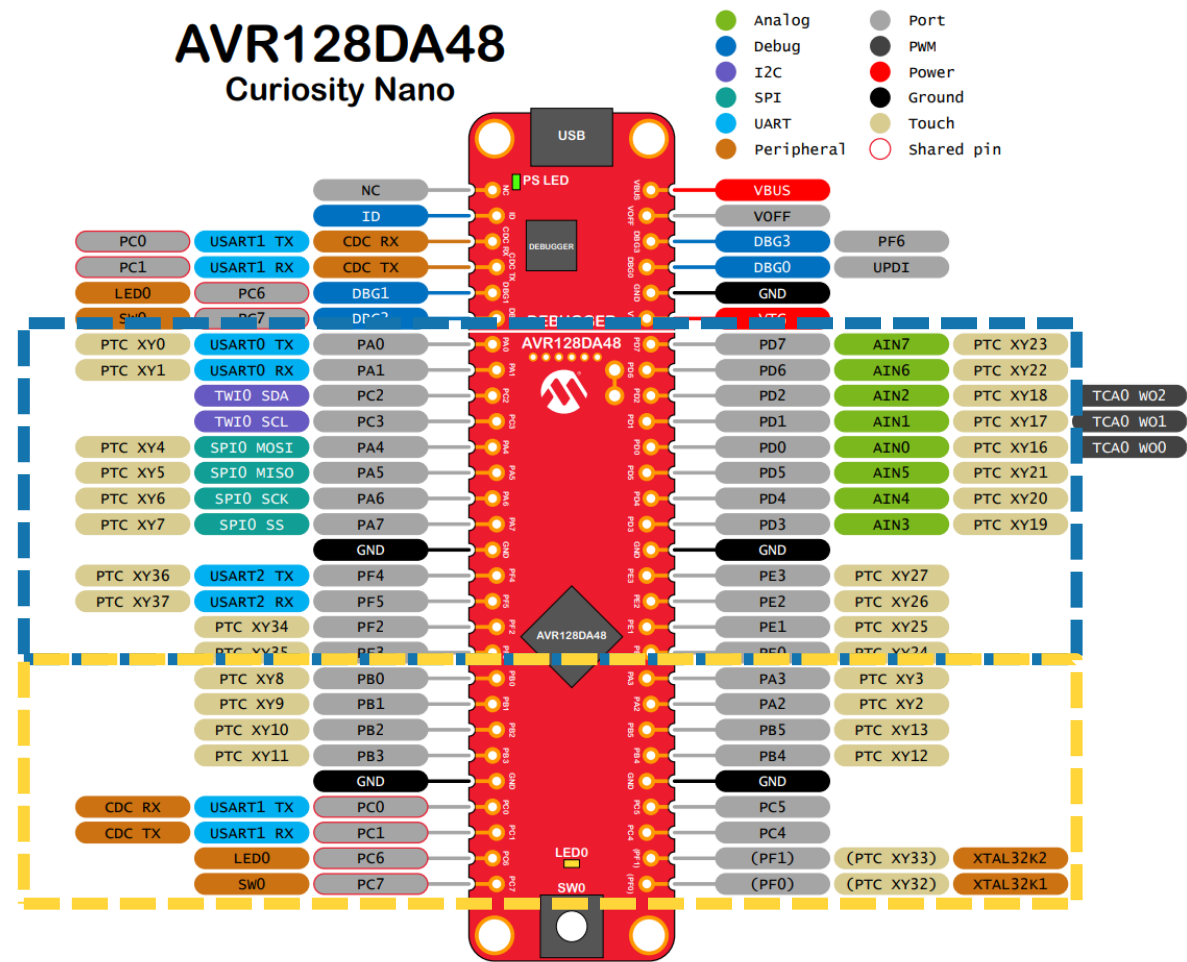


# APP-AII MCU 2023 的方塊圖

All MCUs face to the same Peripherals

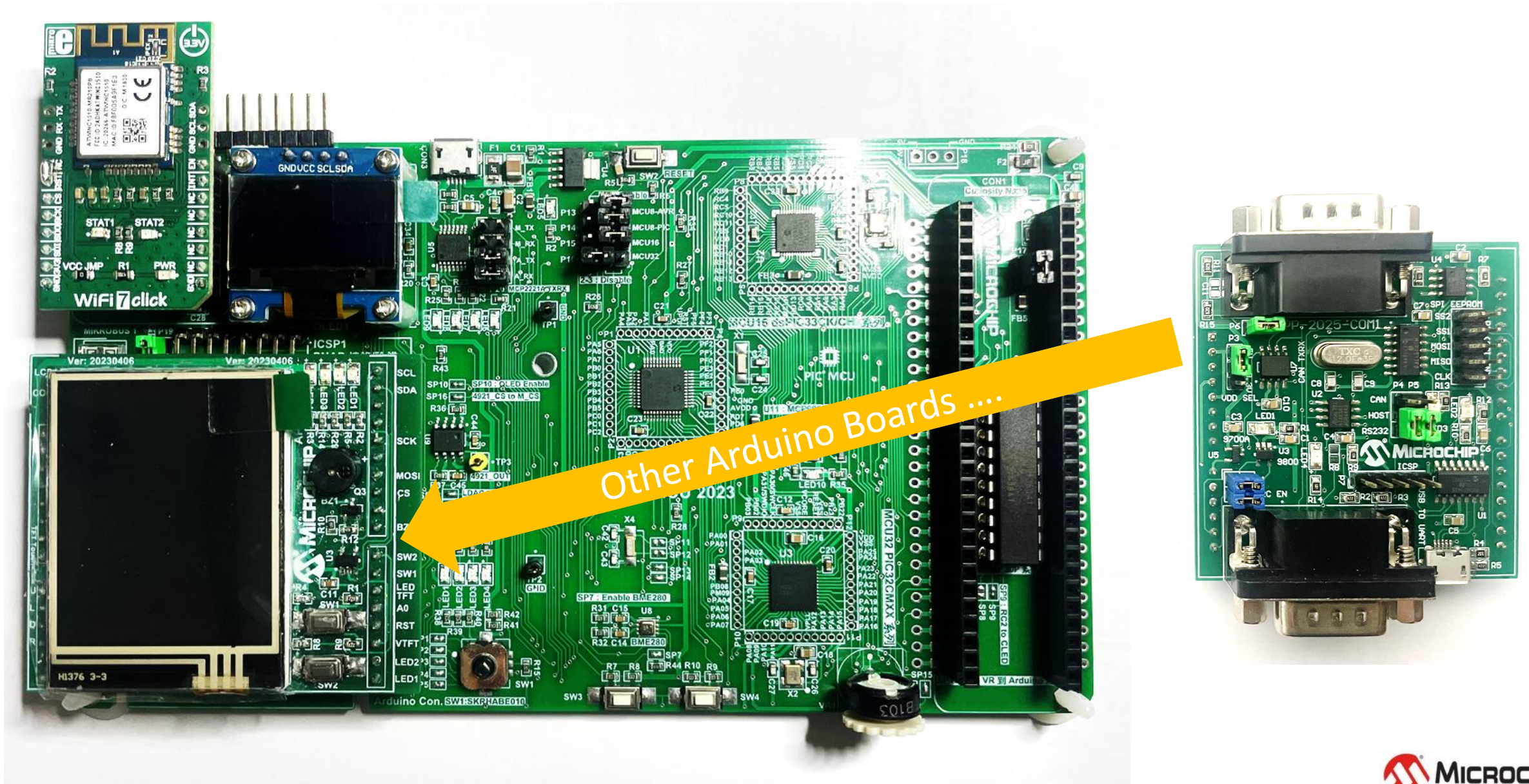


# 參考 Curiosity Nano BASE – 保留 mikroBUS-1，Arduino and on-board 周邊由 mikroBUS-2/3 的信號以及多餘的接腳來控制



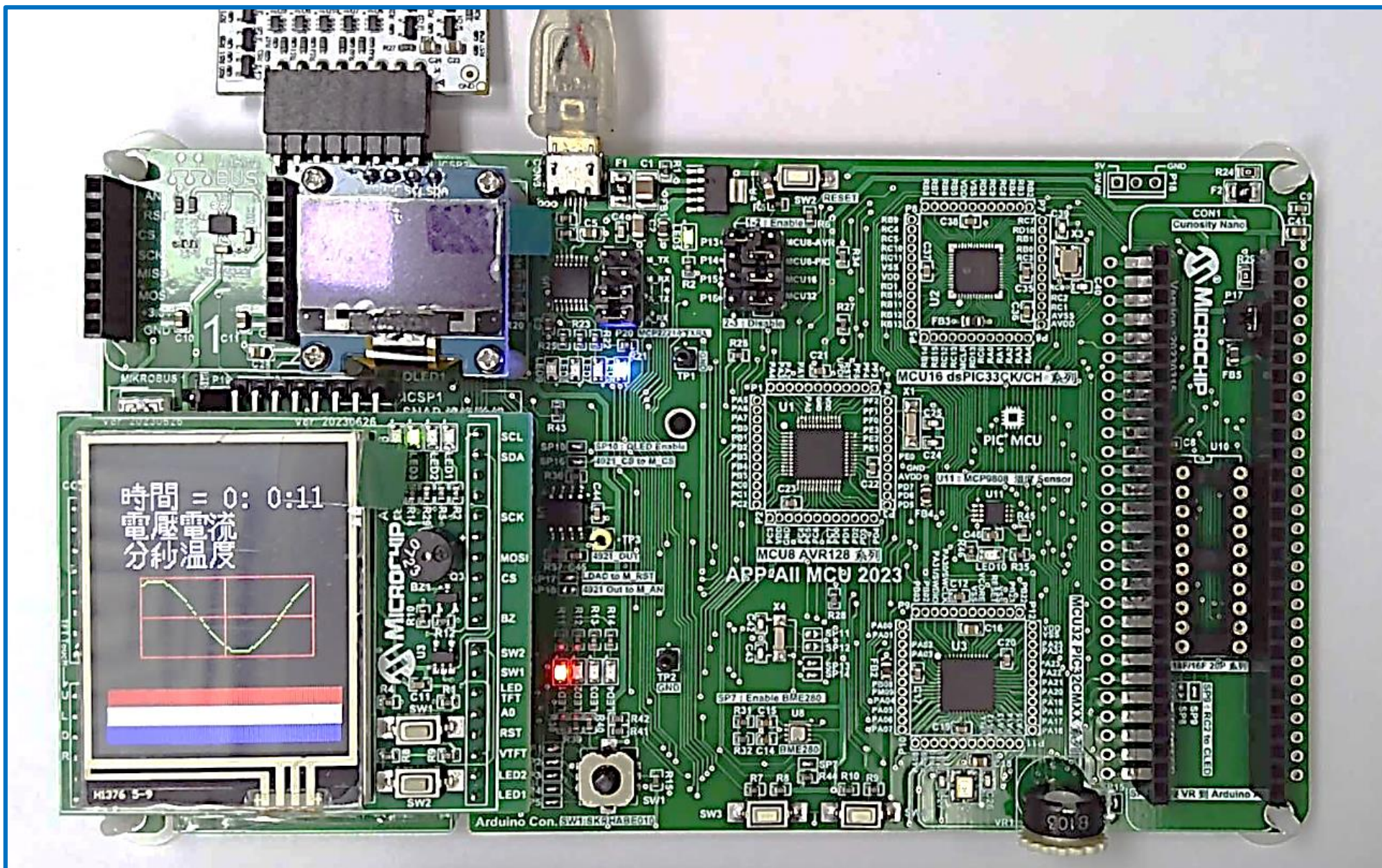


## APP-All MCU2023 保留了同時接 mikroBUS & Arduino 的介面





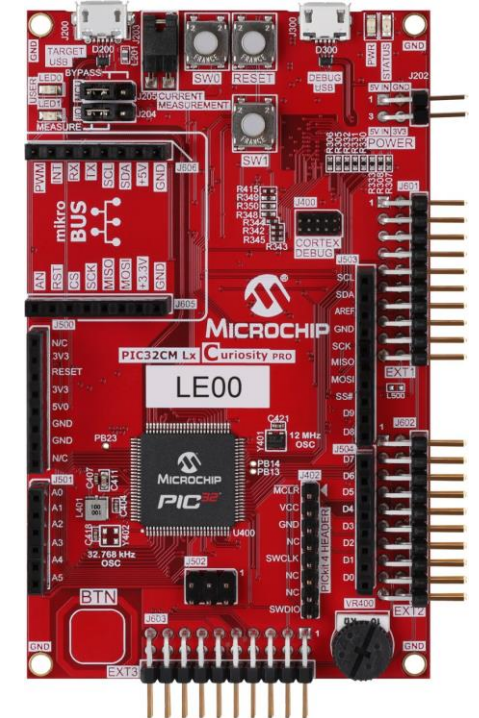
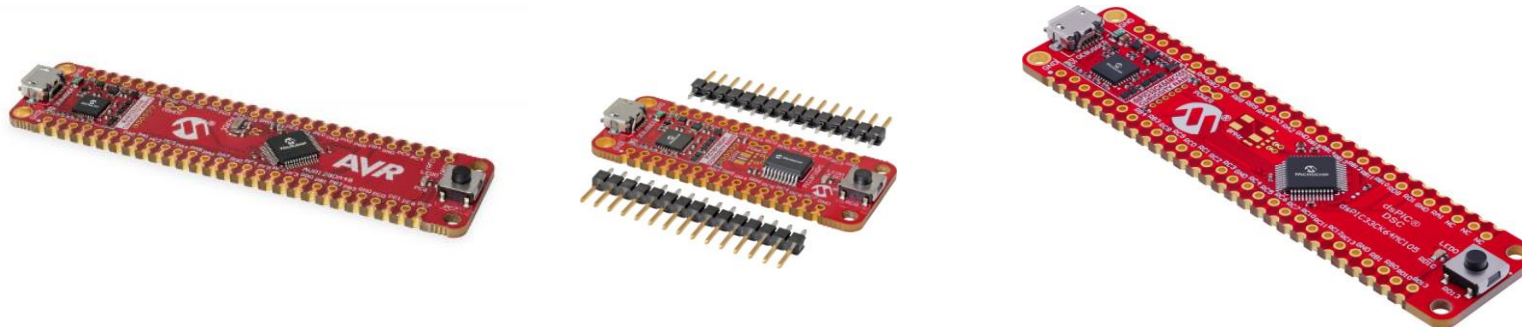
# 使用 Arduino 介面擴充的範例



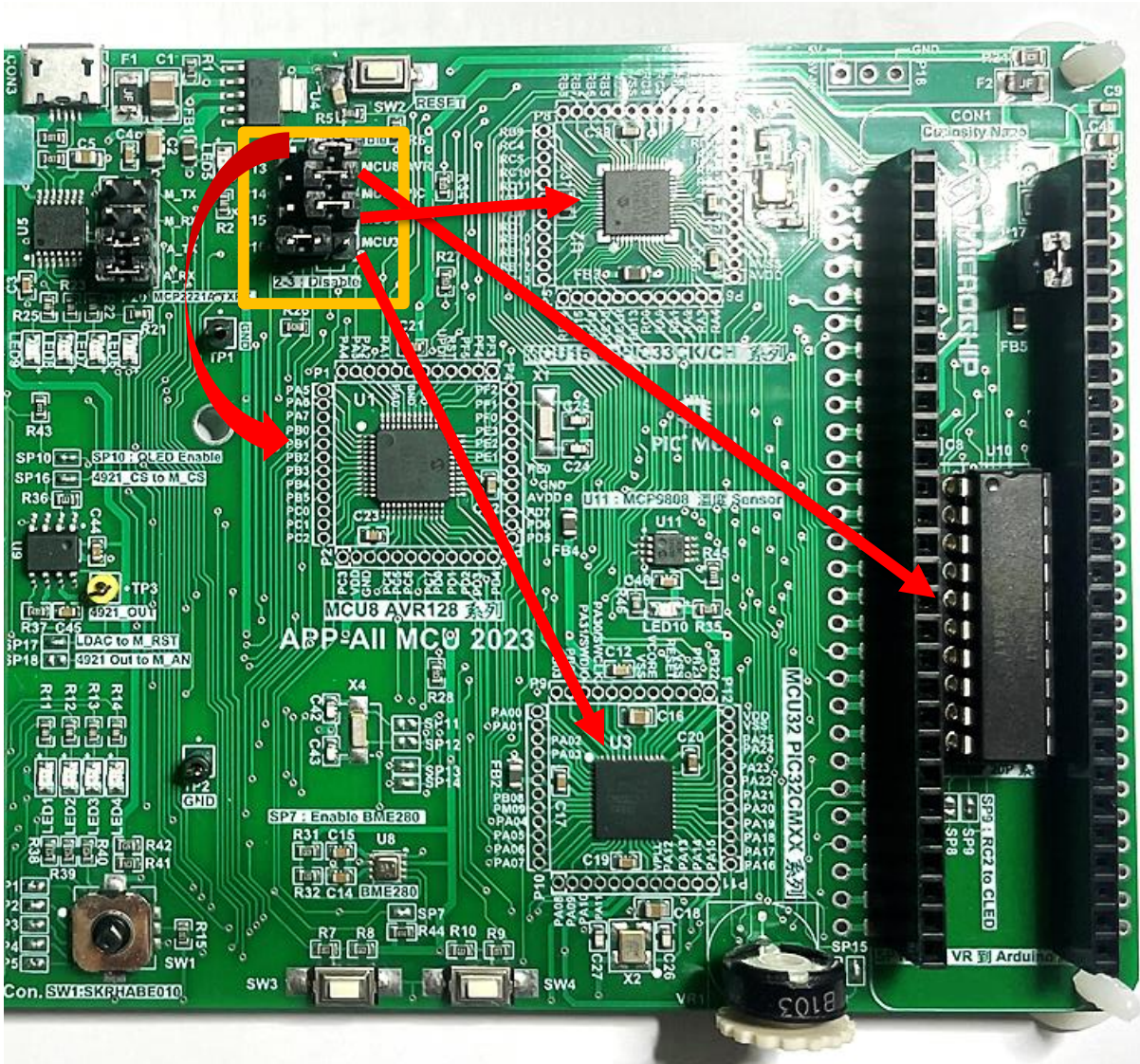


# APP-AI MCU 2023 所安裝的 Microchip MCU

- **MCU8 :**
  - AVR128DA48 – Refer to AVR128DA48 Curiosity Nano - DM164151
  - 20-Pin DIP PIC16F/18F – Refer to PIC18F16Q41 Curiosity Nano - EV26Q64A
- **MCU16 : dsPIC33CH256MP505 :**
  - **Main Core 90 MHz and Secondary Core 100 MHz Operation**
  - Refer to dsPIC33CK64MC105 Curiosity Nano - EV88G73A
- **MCU32 : PIC32CM2532LE0048 : Arm® Cortex®-M23**
  - Refer to PIC32CM LE00 Curiosity Pro Evaluation Kit - EV80P12A
  - Keep the same pin-assignment for mikro BUS 😊



# 使用 P13 ~ P16 來選擇可以工作的 MCU

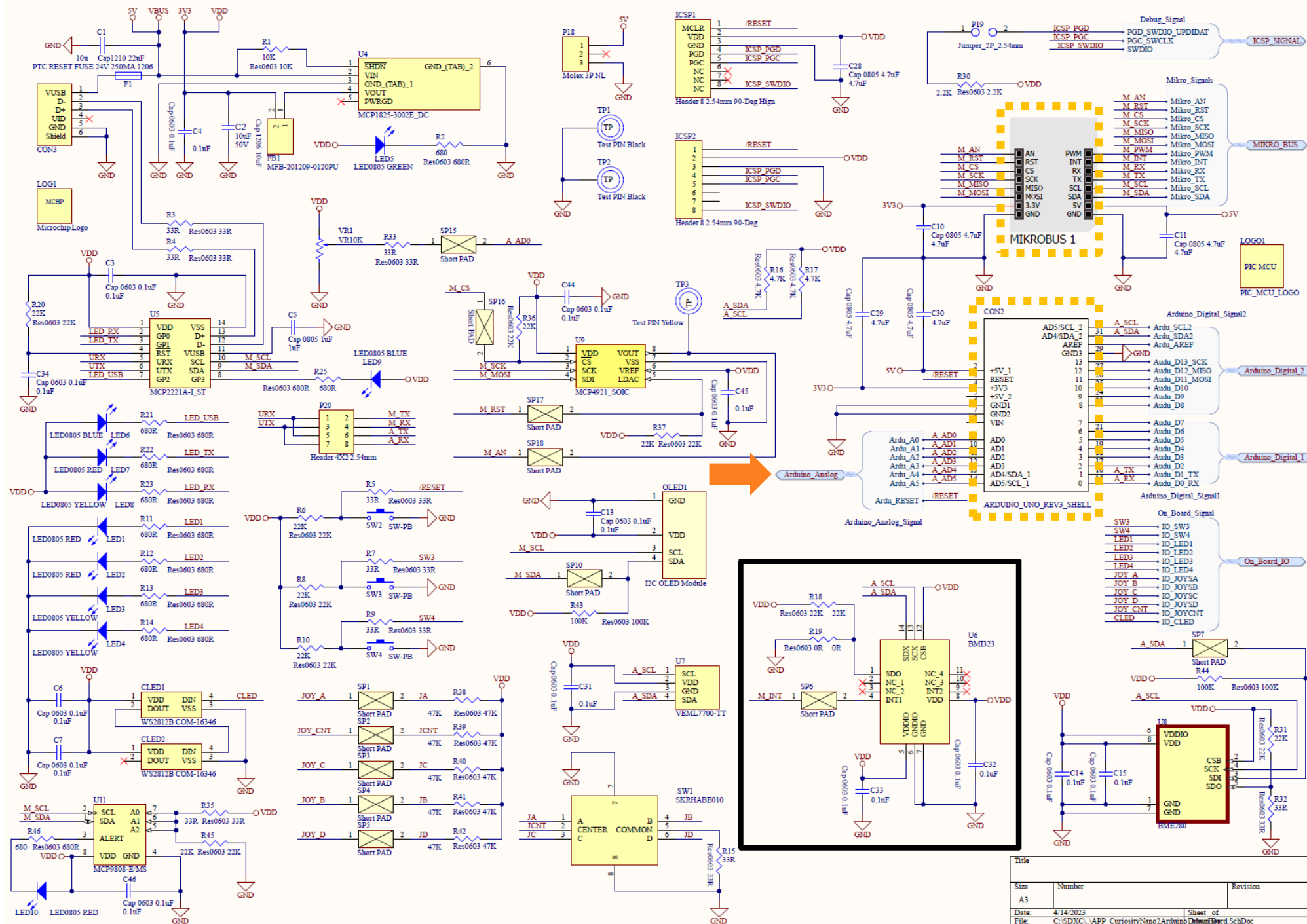


- **P13 :**
  - MCU8 – AVR128DA48
- **P14 :**
  - MCU8 – 20Pin PIC18/PIC16
- **P15 :**
  - MCU16 - dsPIC33CH256MP505
- **P16 :**
  - MCU32 – PIC32CM3532LE000048
- **Pin-1&2 : Reset Pin Release**
- **Pin-2&3 : Reset Pin HOLD**
- 也可以插接其他的 Curiosity Nano EVB 到 CON1 來練習其他 MCU
  - 板上其他 MCU 的 RESET 接腳都要設定在 RESET 狀態
- **Exception : You can Release 2+ MCUs to do multi-MCU interactive programming**
  - i.e. I2C Master/Slave Communication

# APP-All MCU 2023 的線路圖 (For AVR128DA48)

---

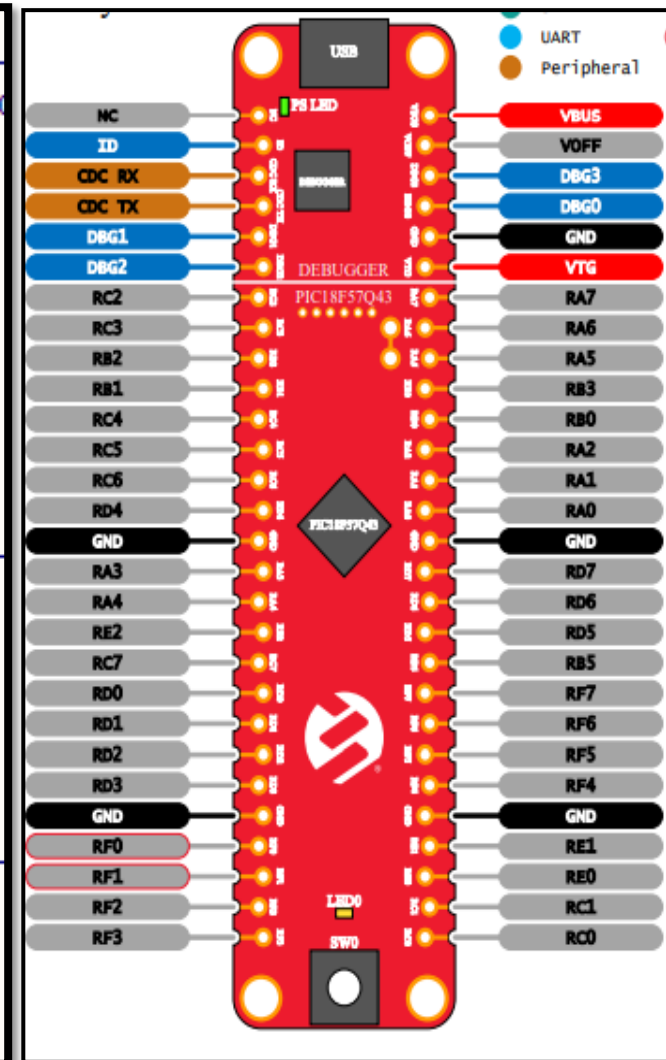
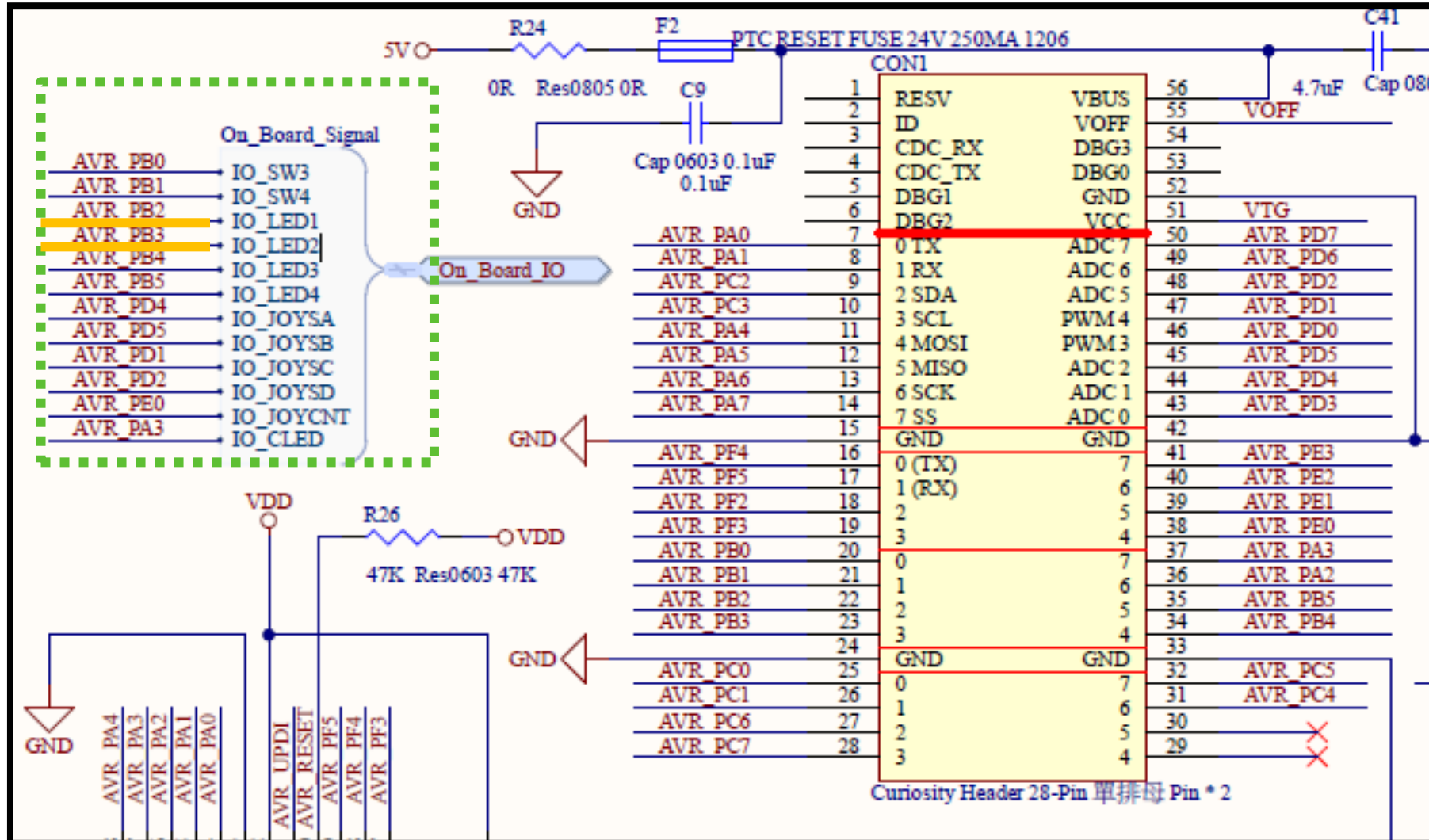






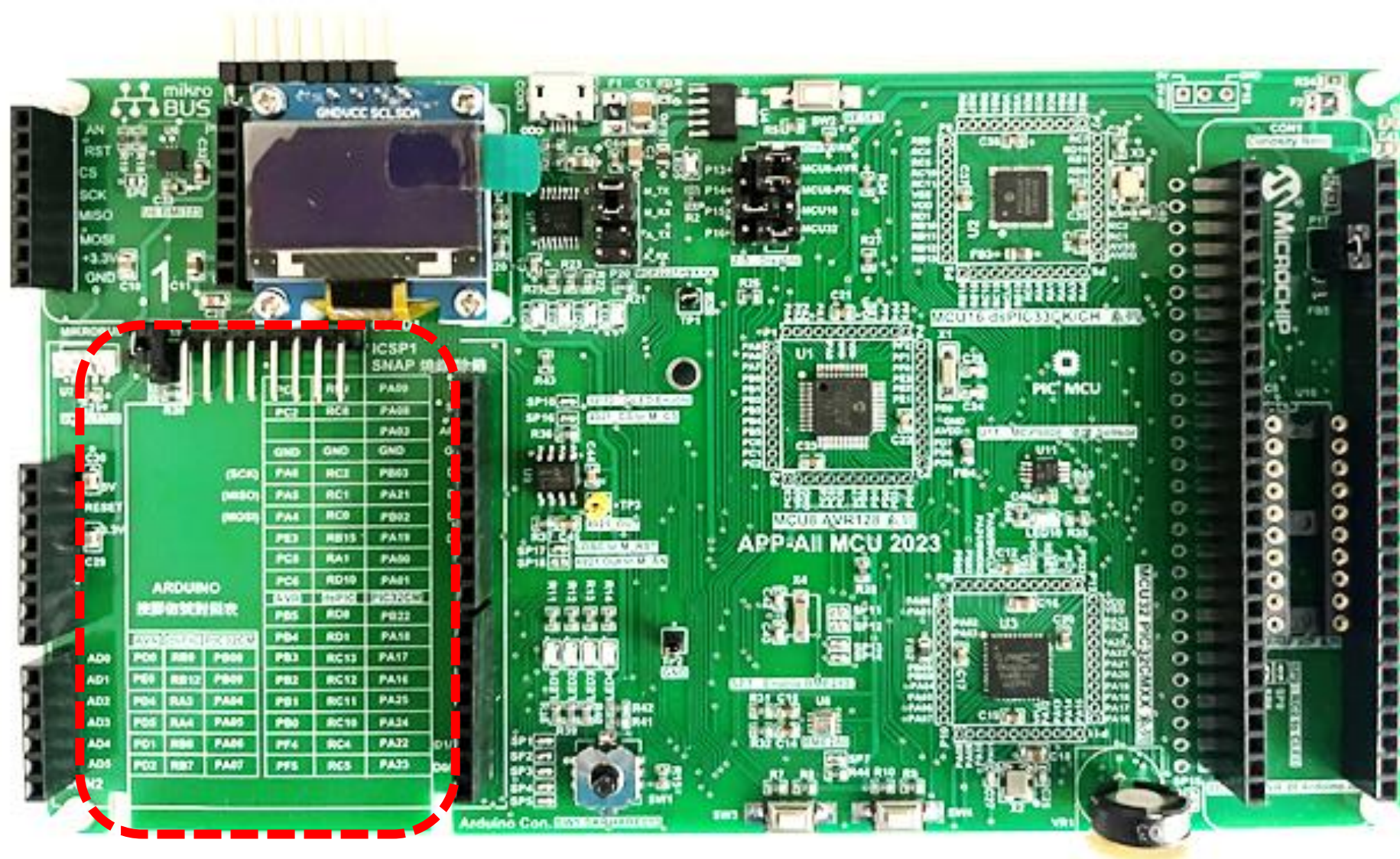
也可以使用其他的 Curiosity Nano (Disable 所有 On-Board 的 MCU

- 對照 CON1 上的 AVR128DA48 排列來找出對應的接腳
- For example ... LED1 = AVR\_PB2 = RD2 @ PIC18F57Q43



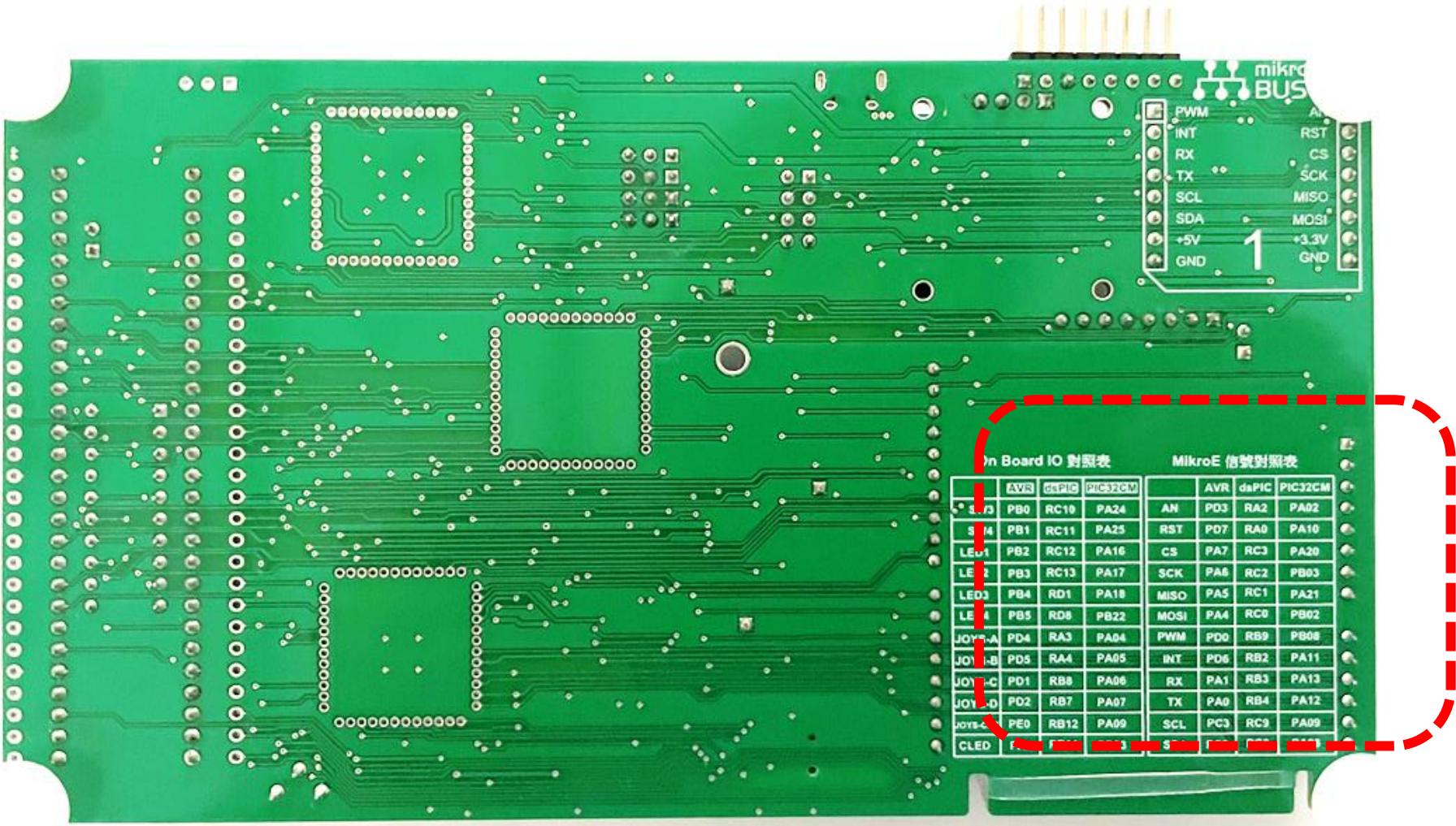


# APP-AII MCU 2023 板上有對 Arduino 介面的接腳說明





# APP-All MCU 2023 has clear Pin Mapping for mikro bus and On-Board I/O too



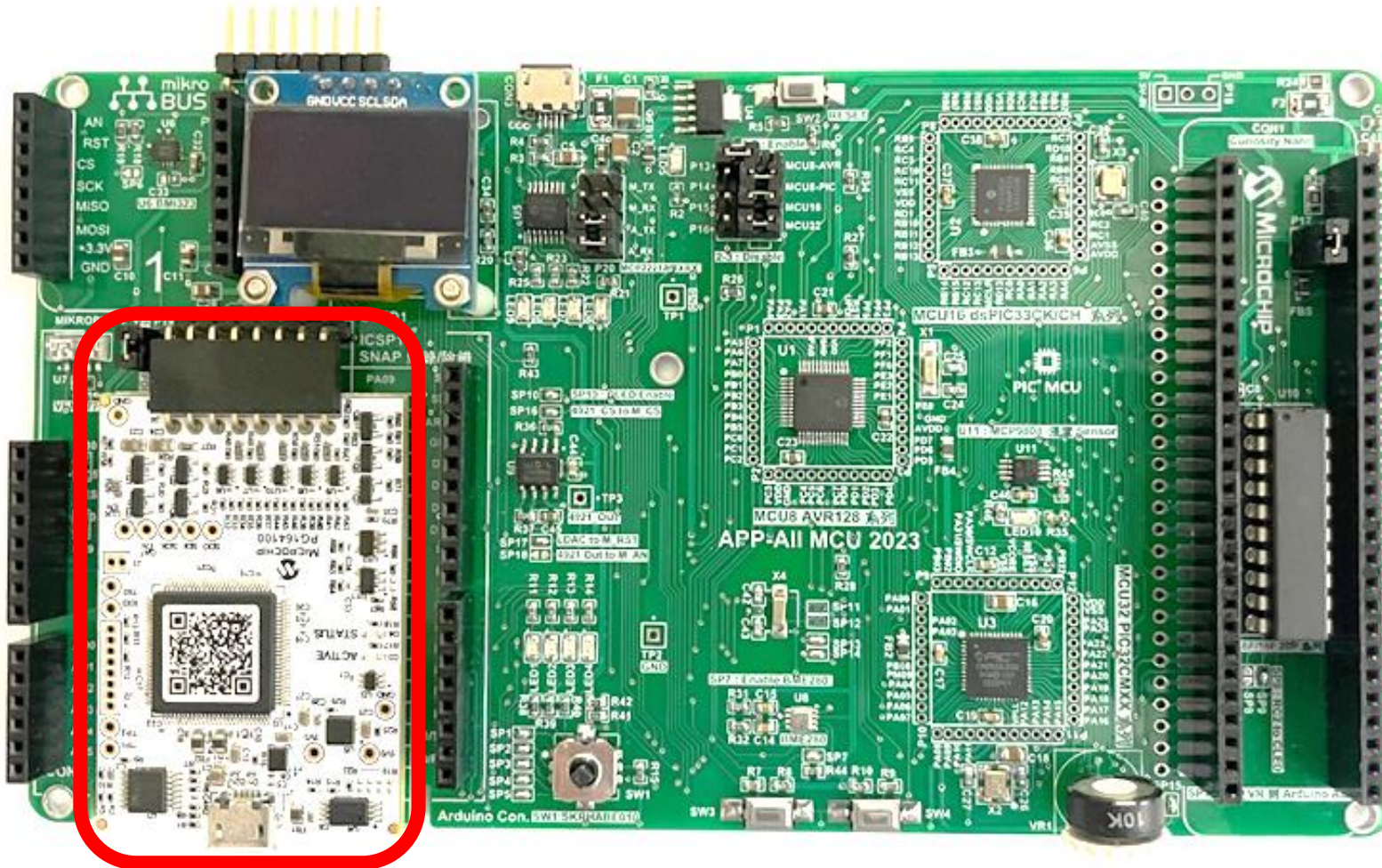


# MPLAB® Snap can be integrated to APP-AI MCU 2023

已經將使用 SNAP 透過 UPDI 燒錄/除錯的必要修改以 **P19** 實現


**ENT36**

[https://ww1.microchip.com/downloads/en/DeviceDoc/ETN36\\_MPLAB%20Snap%20AVR%20Interface%20Modification.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/ETN36_MPLAB%20Snap%20AVR%20Interface%20Modification.pdf)



# Pinout of Debug Interface - SNAP & PICkit4/5

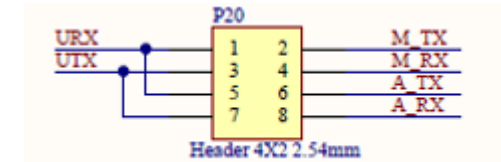
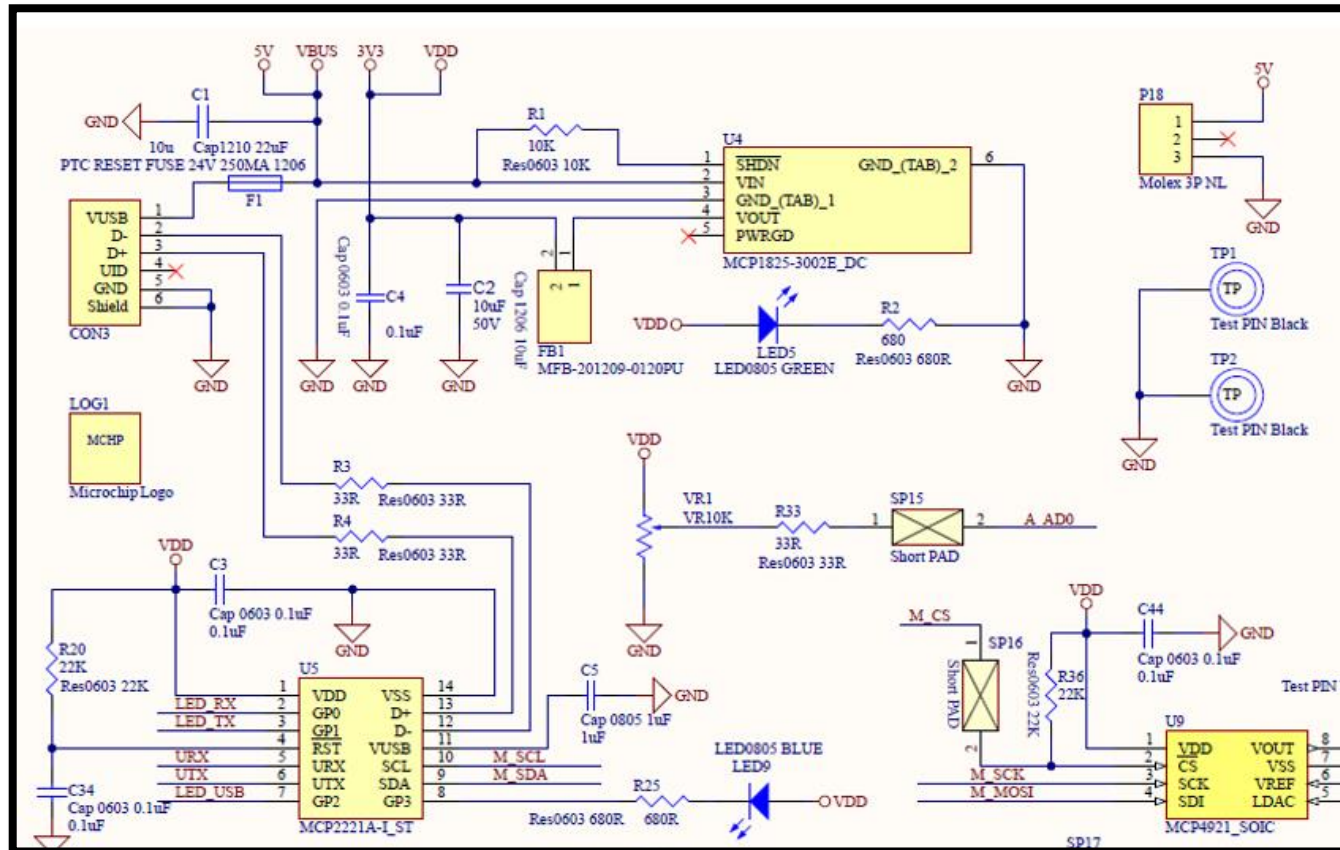
We are talking about Cortex-M0 Family today

MPLAB Snap			DEBUG								
Connector	Pin #	Pin Name	ICSP (MCHP)	MIPS EJTAG	CORTEX® SWD	AVR® JTAG	AVR ISP(&DW)	UPDI	PDI	debugWIRE	TPI
	1	TVPP	MCLR	MCLR	MCLR						
	2	TVDD	VDD	VIO_REF	VTG	VTG	VTG	VTG	VTG	VTG	VTG
	3	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND
	4	PGD	DAT	TDO	SWO	TDO	MISO	DAT	DAT		DAT
	5	PGC	CLK	TCK	SWCLK	TCK	SCK				CLK
	6	TAUX	AUX			RESET	RESET		CLK	dW	RST
	7	TTDI		TDI		TDI	MOSI				
	8	TTMS		TMS	SWDIO	TMS					



# Host Interface – MCP2221A for UART and I<sup>2</sup>C

Full LED Indicator, Connect to both mikro BUS and Arduino connector → **Select by P20**



# APP-AI MCU 2023 development resource @ MCHP TW

← ↻ 🏠 <https://www.microchip.com.tw/index.php> 🔍 ⚙️ ☆

microchip.com Microsoft Office 首... MICROCHIP in Digit... Bookmarks Microchip InfoDepot Sign in to GitHub ... 3D ContentCentral... GC GrabCAD: Design C... Pyth

了解更多

**MICROCHIP** 🏠 回首頁 ▾ 主選單 ▾ Webinar資料區 CAE專家教室 技術問與答 ▾

**搜尋** 🔧

請輸入關鍵字 🔍


進階搜尋

**會員選單** 🔧

- 🔧 管理區
- 👤 檢視帳號
- ✎ 編輯帳號
- 🔔 通知
- 📧 收件箱 1
- 🚪 登出

**主選單** 🔧


**研討會/eRTC** 🔧



APP All MCU 2023 開發資源下載

嵌入式解決方案研討會

Sustainability  
發掘生態永續共存的商機




📍 台北 5月11日 | 台大集思會議中心

Microchip 2023 春季 ESS - 台北場 - 點我

嵌入式解決方案研討會

Sustainability  
發掘生態永續共存的商機




📍 新竹 5月10日 | 豐邑喜來登大飯店

Microchip 2023 春季 ESS - 新竹場 - 點我

嵌入式解決方案研討會

Sustainability  
發掘生態永續共存的商機




📍 台中 5月9日 | 長榮桂冠酒店

Microchip 2023 春季 ESS - 台中場 - 點我


嵌入式解決方案研討會

Sustainability  
發掘生態永續共存的商機



📍 高雄 5月8日 | 蓮潭國際會館

Microchip 2023 春季 ESS - 高雄場 - 點我

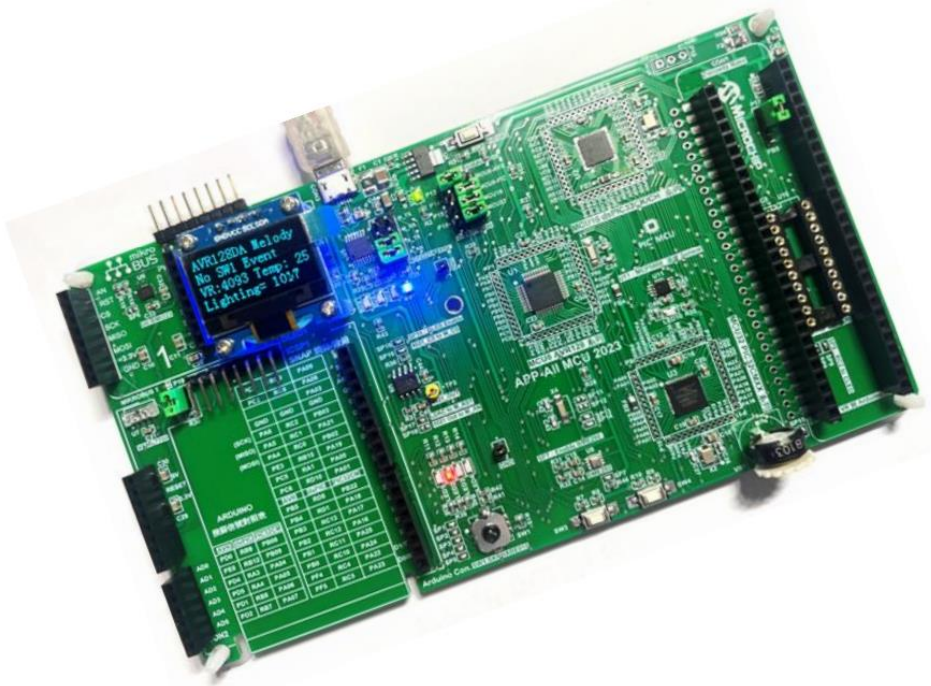


APP-SAM9X60 Hobbv Kit 開發資源連結



# APP-AII MCU 2023 開發資源下載的頁面(持續補充中)

## APP-AII MCU 2023 實驗板開發資源介紹



APP-AII MCU 2023 是 Microchip Taiwan Office 的另外一個嘗試，旨在於提供簡單並實用的開發工具，讓廣大的 Microchip 之友們可以不斷地依照自己、公司、客戶及產業的需求來對所從事的設計或專案來做升級或新功能的導入。

Microchip 的 MCU 產品在業界一直廣受好評，涵蓋的範圍及為廣泛，不但有完整的 8、16、32 位元的 MCU 並且有多種不同的核心讓開發者可以根據需要來做最適切的選擇。

在 APP-AII MCU 2023 上面您可以搭配豐富的 On-Board 周邊，對 Microchip 的 8、16、32 位元 MCU 做出客觀而且正確的評估。以下為開發資源的連結，我們也會持續地更新。

- [APP-AII MCU 2023 實驗板簡介](#)
- [APP-AII MCU 2023 線路圖](#)
- [APP-AII MCU 2023 實驗板出廠 HEX 檔案](#)

# AVR<sup>®</sup> DA Product Family

## 重要的周邊及功能介紹



---

A Leading Provider of Smart, Connected and Secure Embedded Solutions



SMART | CONNECTED | SECURE

# AVR128DA48

**New Functional Safety Ready AVR®  
Microcontroller Family Enables Real-Time  
Control, Connectivity and HMI Applications**

Next-Generation AVR DA MCUs Features Core Independent  
Peripherals, Advanced Analog and On-Chip Communications

[www.microchip.com/avr-da](http://www.microchip.com/avr-da)



# Oscillator and Clocks

- **High-precision internal oscillator**
  - Selectable frequency up to 24 MHz with auto tuning option using 32 kHz crystal
  - +/-2% accuracy across 0 – 70°C without auto tuning
  - XTAL free serial communication
- **PLL up to 48 MHz for high frequency operation of peripherals**
  - For high speed PWM using Timer D
- **Internal ultra low power 32kHz RC oscillator**
  - User configurable with +/-2% accuracy
  - Ability to calibrate from the high-speed internal RC oscillator
  - Can be used as main clock source
- **External 32kHz crystal oscillator**
  - Used for the real-time-clock operation
- **External clock input**
  - Ability to clock the device from an external clock source



# Ex1 : AVR128DA48

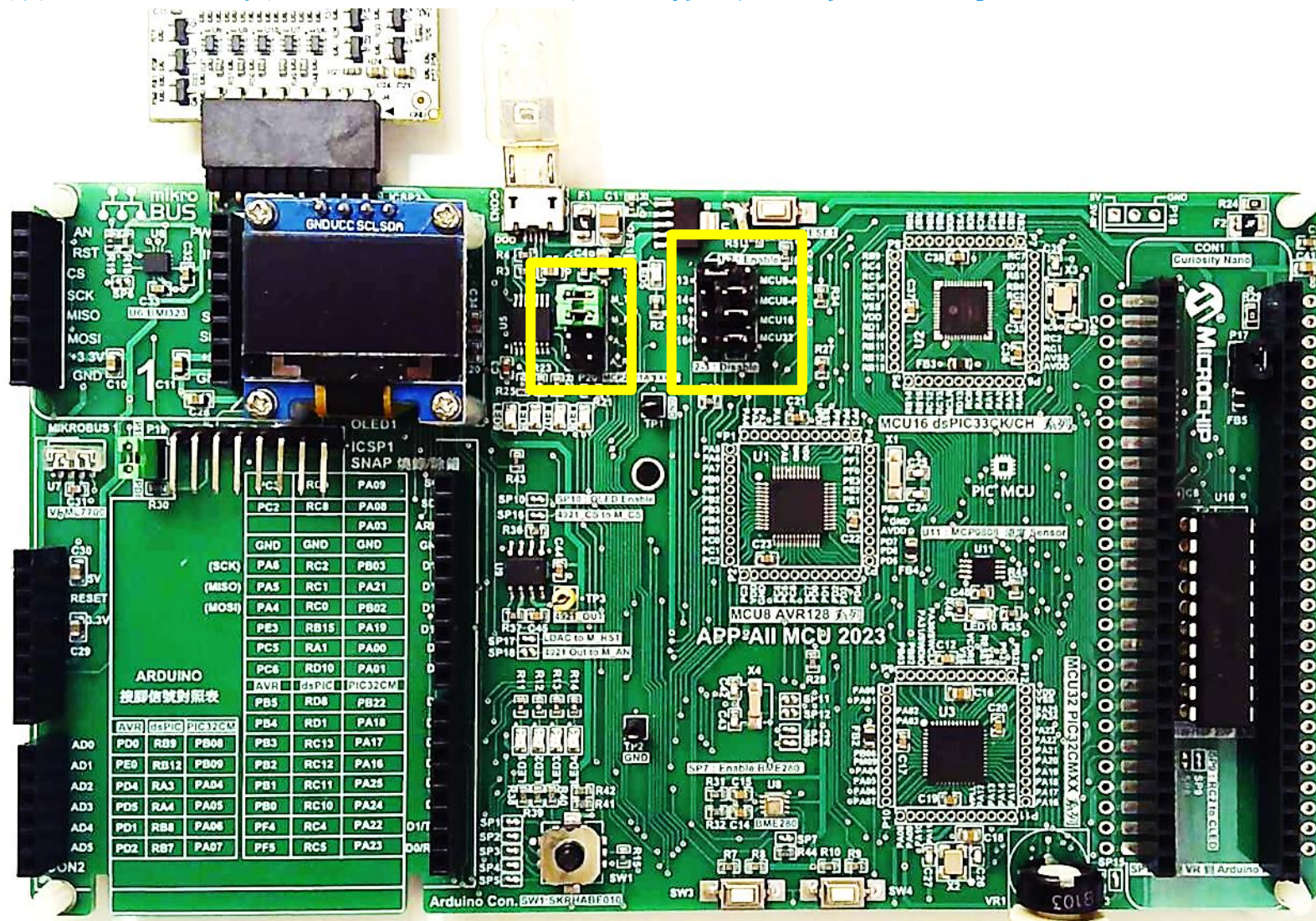
## Basic I/O

---

Use : MPLAB XIDE V6.10

# 做實驗前的裝備檢查

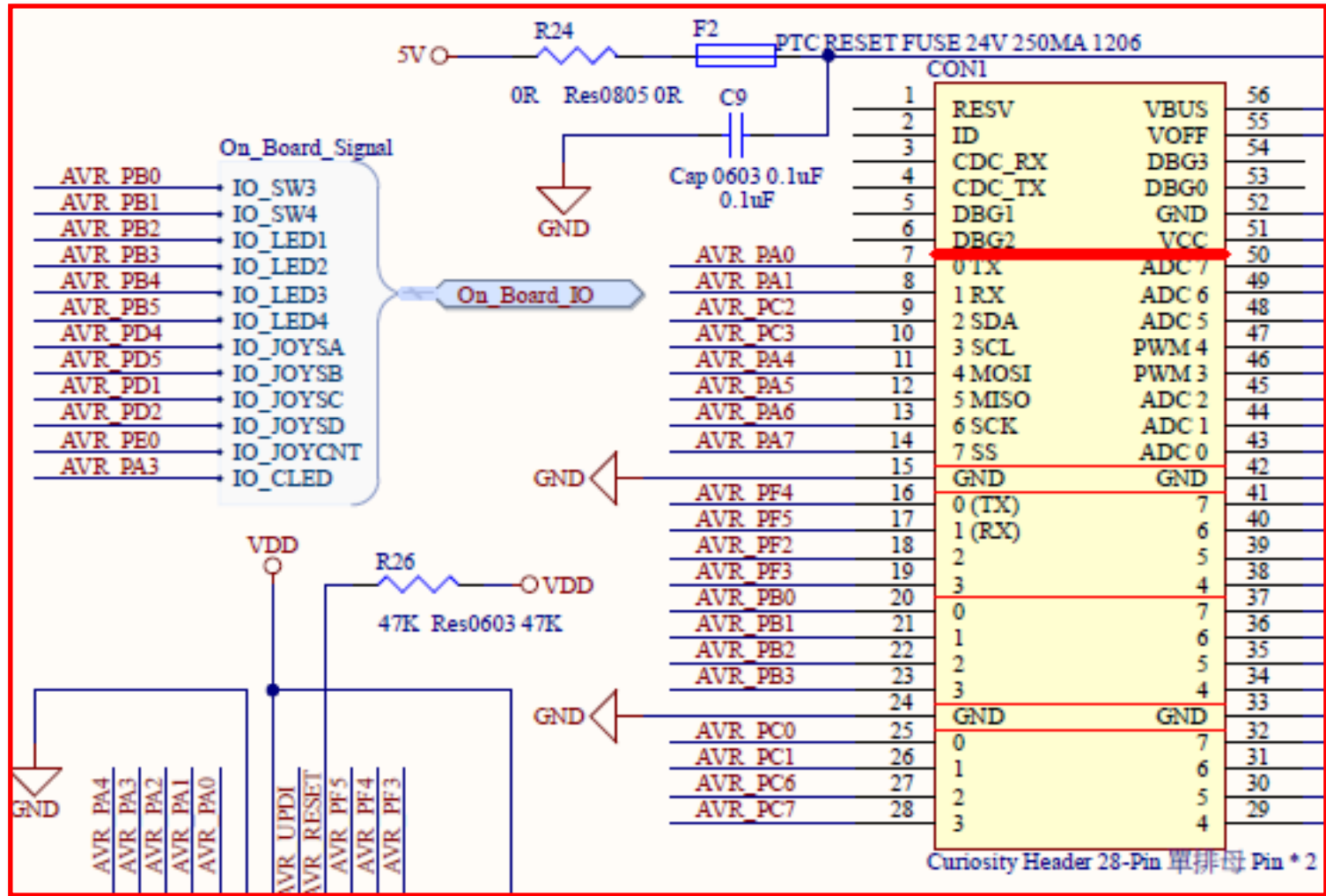
請確定硬體已經做了適當的連接與設置 (P13~P16, P20)





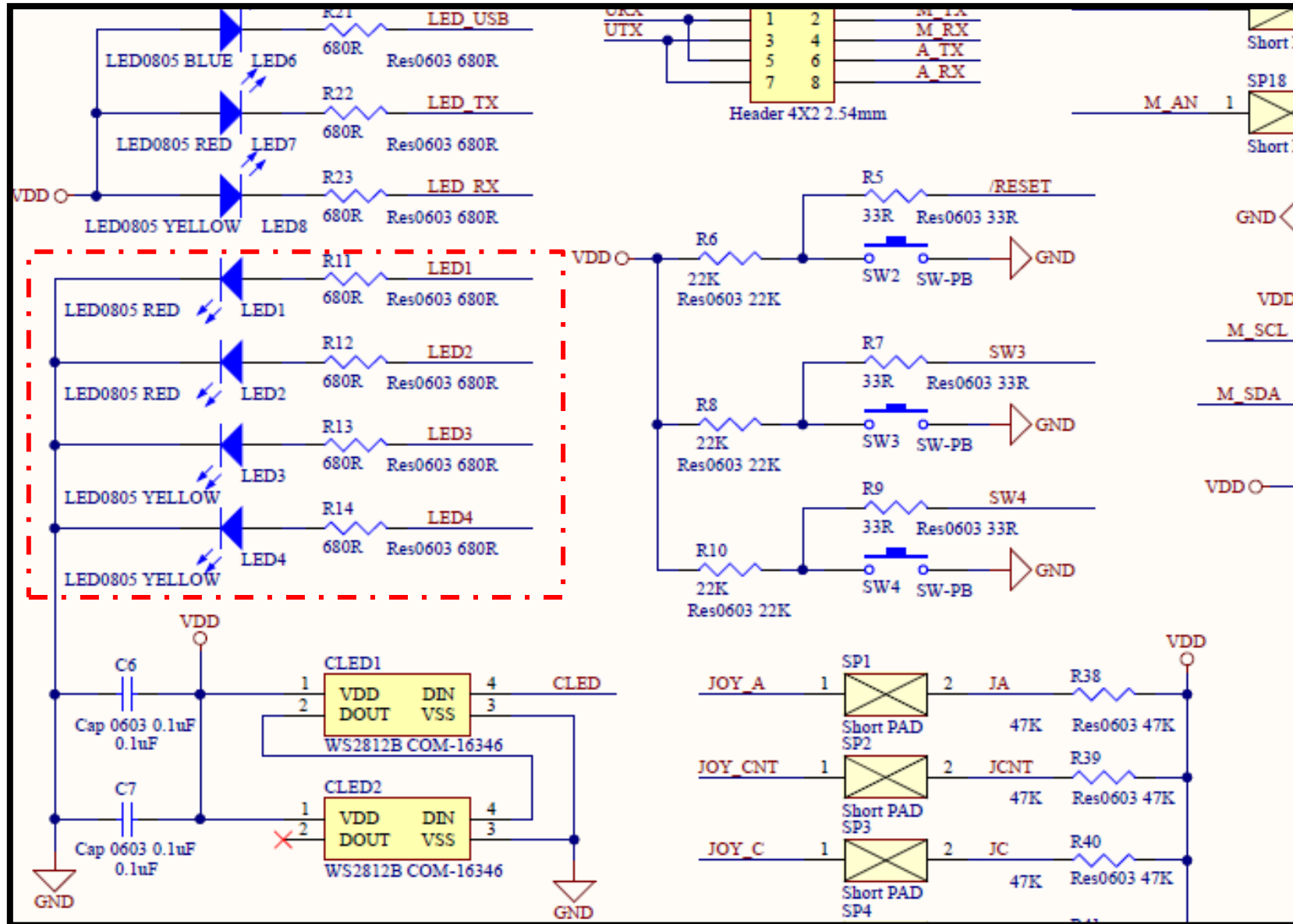
# 練習一 Project Guide

- **Project Name :**  
**Lab\_AVR128DA48**
- **Project Folder :**
  - RTC\AVR101
- **使用的資源**
  - AVR128DA48 控制 LED1 ~LED4
    - LED1 : AVR\_PB2
    - LED2 : AVR\_PB3
    - LED3 : AVR\_PB4
    - LED4 : AVR\_PB5
  - Melody 提供的 DELAY Driver
- **Ex1 要達成的功能**
  - 完成一個每 200 ms 位移一次的跑馬燈
  - 位移的方向由 LED1 住次向 LED4 , 並且在盡頭轉換方向



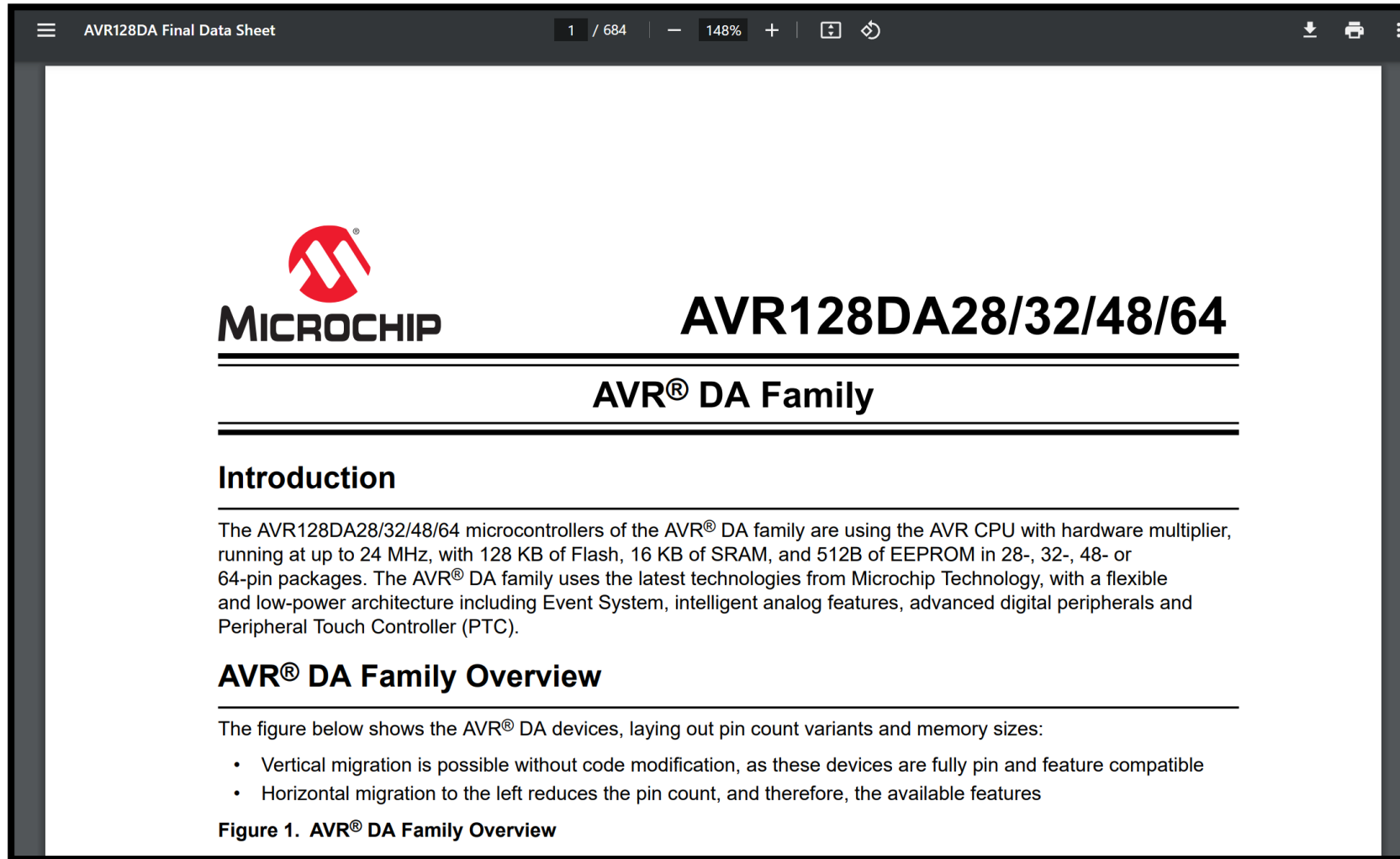


# LED1 ~ LED4 為 High Driven，輸出為 1 時燈亮



# Why MCC Melody or MCC Classic ?

- AVR 有多樣的功能、豐富的內建周邊、手冊多達 684 頁 ☹️
- 避免因為人為疏忽而沒有對周邊做完整的初始化



# 您可以在 Microchip 網站尋找開發資源

## >> Tools & Software 、 Solution 、 Product

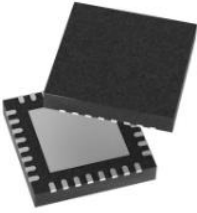
- <https://www.microchip.com/en-us/education/developer-help>

## Developer Help

Microchip's technical training solution for all your needs

Developer Help provides short self-paced technical training and reference material to get you answers when you need them. Choose from a wide variety of training materials that clearly explain Microchip's products and applications. Topics include development tools, software development, MCUs and MPUs, wired and wireless connectivity, analog, FPGAs, touch control, motor control, power supplies and much more.


All of our training on Developer Help is self-paced and available 24 hours a day, seven days a week. We strive to provide you with the right training, at the right time, and in the right place to meet all your needs.



### Learn Products

Learn what our products do, how their features work and how to configure them for your applications.


[Learn More](#)



### Learn Solutions

Learn from application examples, code samples and how to configure our products to work in your application.

[Learn More](#)



### Learn Tools and Software

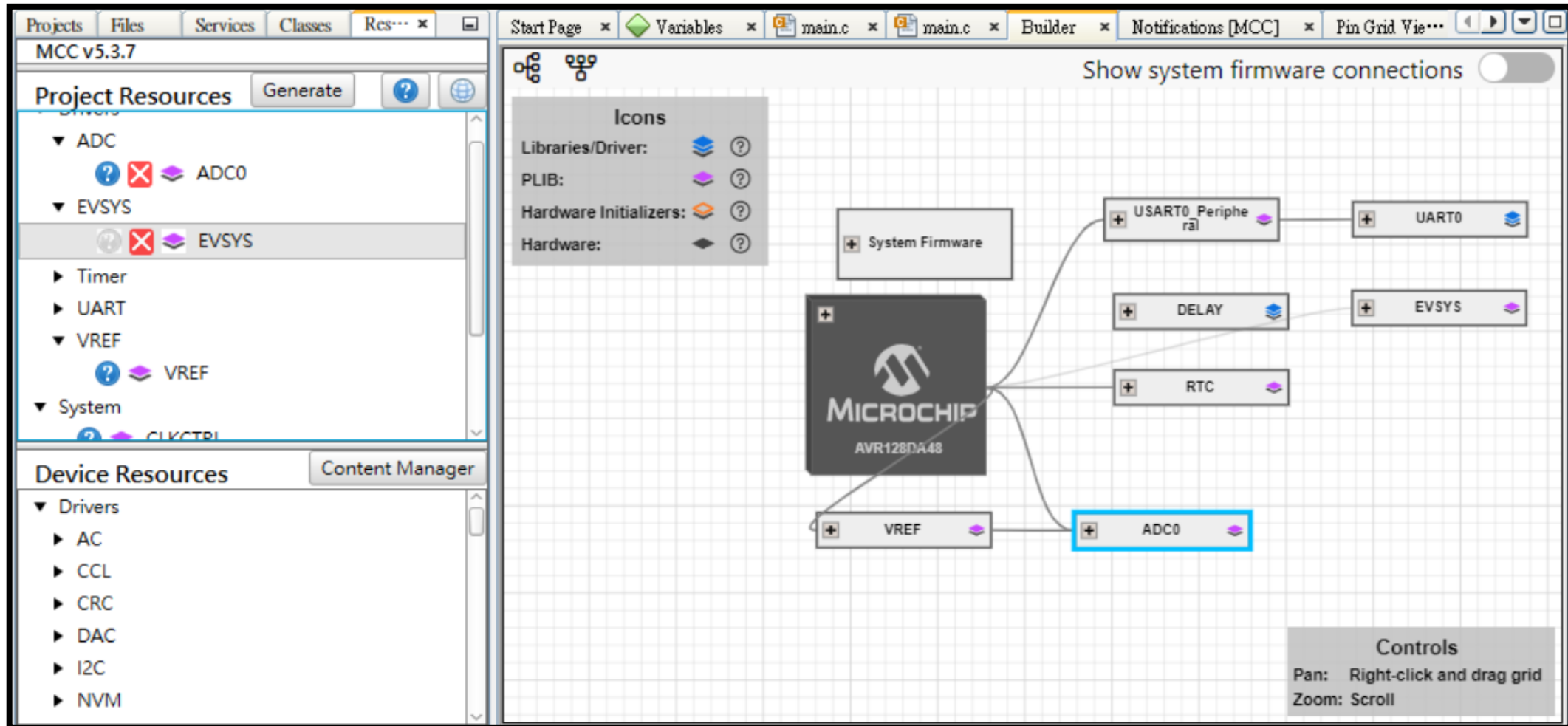
Learn how to use our development tools, both hardware and software, how to install them and how to get the most out of them when developing your application.

[Learn More](#)



# 如何快速找到MCC Melody API 說明文件？(1)

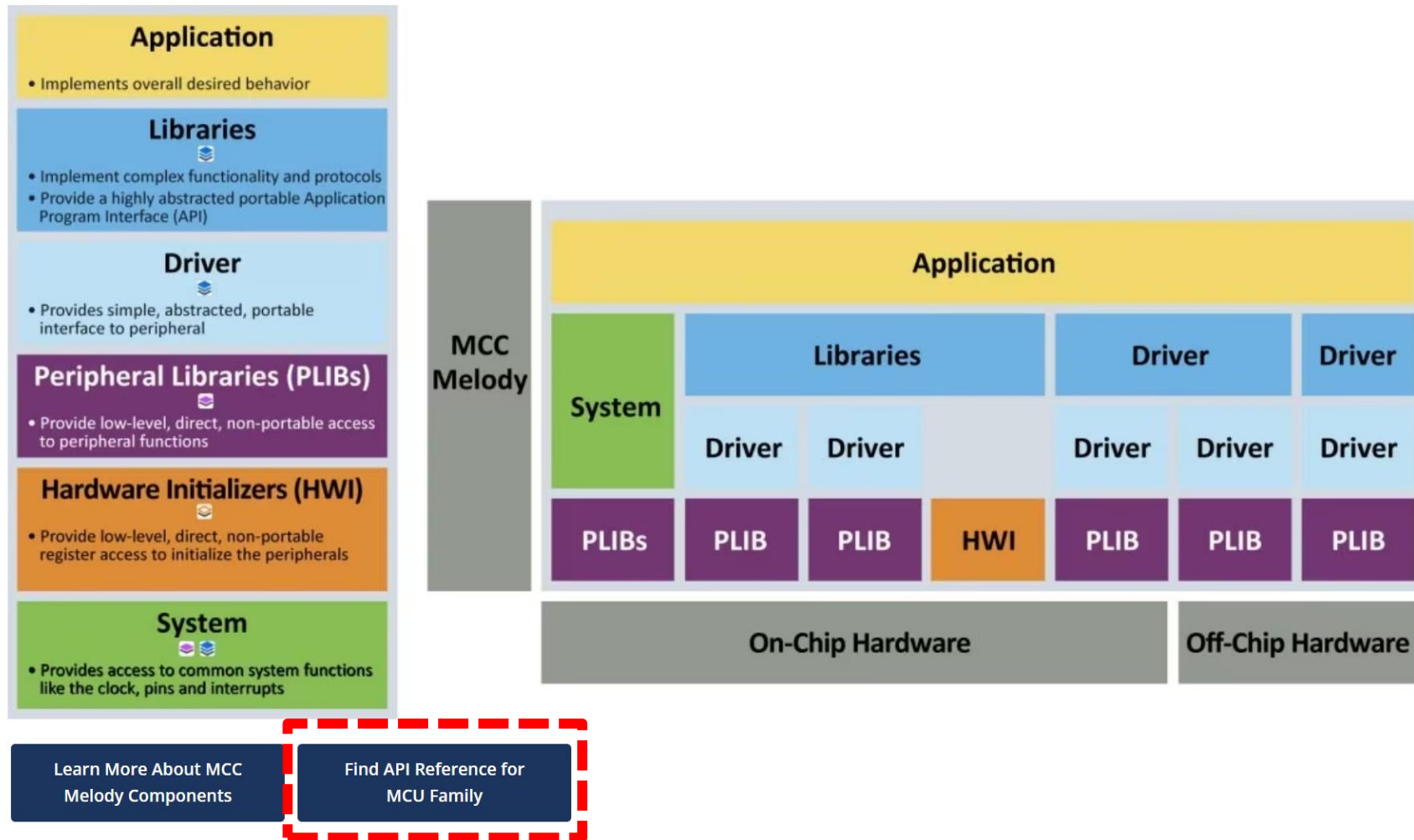
- 在 MCC Melody 中只要點擊  就會直接導向說明文件 (線上)





# 如何快速找到MCC Melody API 說明文件？(2)

- 在 MCC Melody 的專用頁面上找到並點選 “Find API Reference for MCU Family”
- <https://www.microchip.com/en-us/tools-resources/configure/mplab-code-configurator/melody>



# 如何快速找到MCC Melody API 說明文件？(3)

- 直接連上：<https://onlinedocs.microchip.com/oxy/GUID-5A03F818-B7FC-4062-9792-57D08543B586-en-US-7/index.html>

1 MCC Melody Introduction
2 Getting Started with MCC Melody Video and hands-on training to quickly get you up and running with MCC Melody.
> 3 MCC Melody Components
4 Glossary of Terms
5 MCC Content Manager
6 Key Differences Between MCC Classic and MCC Melody
> 7 MCC and MCC Melody API References Individual API References, by MCU family and MCC User's Guide.
> Microchip Information

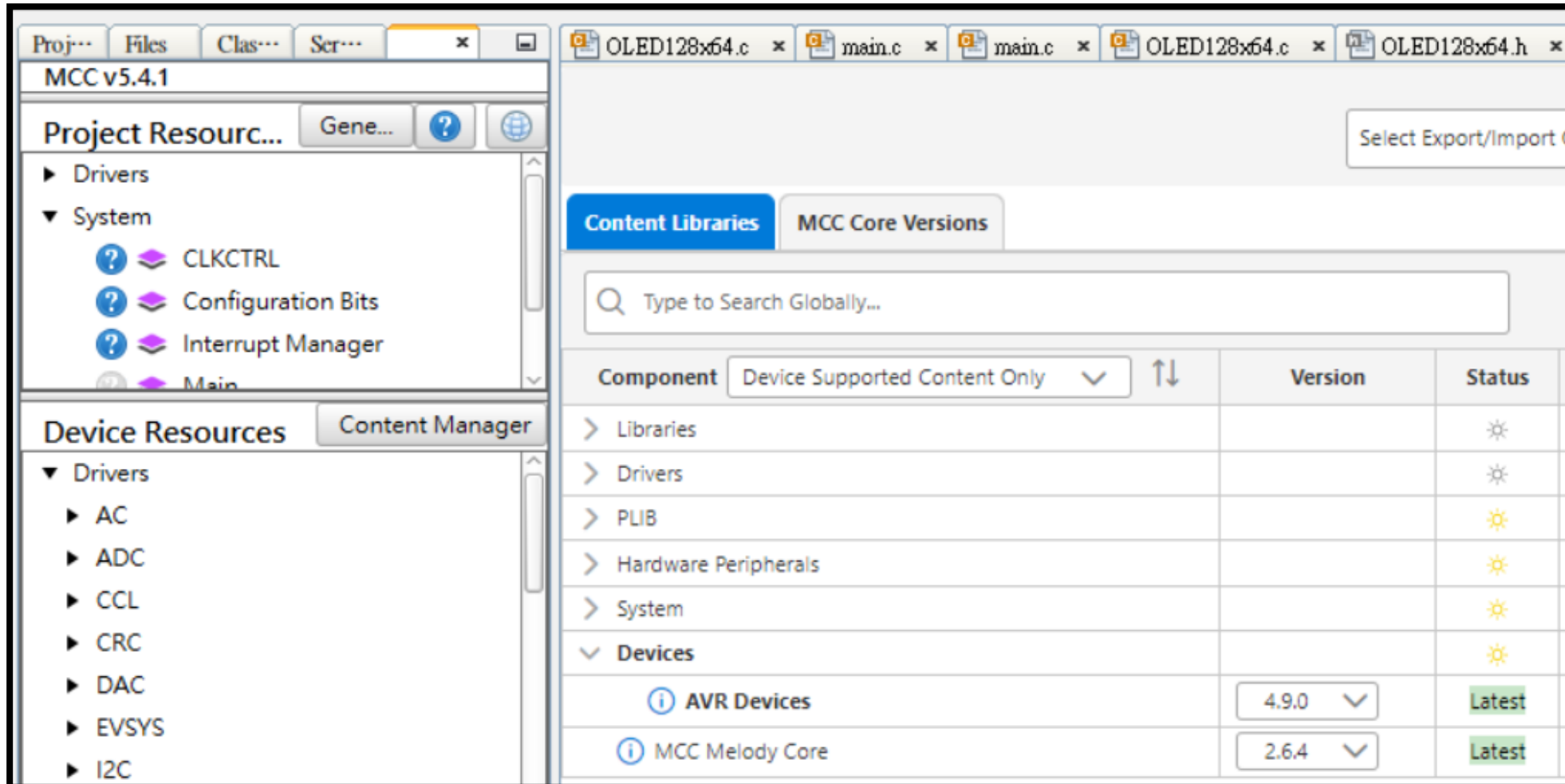
# RESET Pin : 建議設定為 Reset mode

- 因為要與 APP-AI MCU 2023 的其他 MCU 做隔離

The screenshot displays the Microchip MCC v5.3.7 software interface. On the left, the 'Project Resources' and 'Device Resources' panels are visible. The central workspace shows a block diagram of the AVR128DA48 MCU with various peripherals connected. On the right, the 'Configuration Bits' panel is expanded, showing settings for BOD Sample Frequency, BOD Operation in Sleep Mode, BOOTSIZE, CODESIZE, OSCCFG, and SYSCFG0. The 'Reset Pin Configuration' option is highlighted with a red dashed box and is set to 'Reset mode'.

# MPLAB X IDE 、MCC & Content Manager

- MPLAB X IDE V6.15 with Latest version Melody Core & AVR Device 可以正確地完成 AVR-101 的練習





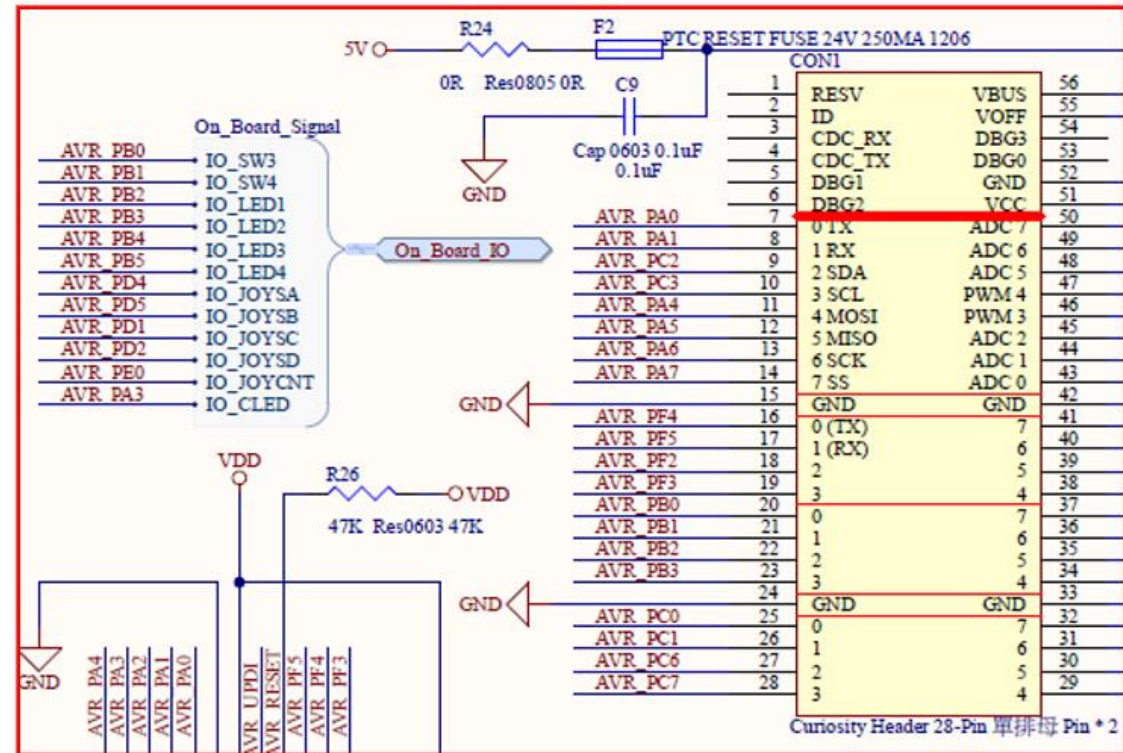
# Lets' go Ex1 by following the procedures in Video 😊

<https://youtu.be/kvxlkxl2sTw>



## 練習一 Project Guide

- Project Name :  
Lab\_AVR128DA48
- Project Folder :
  - RTC\AVR101
- 使用的資源
  - AVR128DA48 控制 LED1 ~LED4
    - LED1 : AVR\_PB2
    - LED2 : AVR\_PB3
    - LED3 : AVR\_PB4
    - LED4 : AVR\_PB5
  - Melody 提供的 DELAY Driver
- Ex1 要達成的功能
  - 完成一個每 200 ms 位移一次的跑馬燈
  - 位移的方向由 LED1 住次向 LED4 , 並且在盡頭轉換方向



# Ex1 的 MCC Project Setting - CLKCTRL

The screenshot displays the MPLAB IDE interface for configuring the CLKCTRL module. The left sidebar shows the Project Resource tree with 'CLKCTRL' selected under the 'System' category. The central workspace contains a grid with components like 'System Firmware' and 'DELAY'. The right-hand panel shows the 'CLKCTRL Settings' with various configuration options. The 'Oscillator Frequency Selection' is highlighted in red, showing '24 MHz system clock'. The bottom section shows the 'Pin Grid View' for the QFN48 package, with pins 44-48 and 43-47 assigned to the CLKCTRL module.

**CLKCTRL Settings**

- Main Clock (Hz): 24000000
- Clock Selection: Internal high-frequency oscillator
- Internal Oscillator Frequency: 1-32MHz internal oscillator
- Oscillator Frequency Selection: 24 MHz system clock**
- PLL Enable: ☐
- Multiplication Factor: PLL is disabled
- External Clock Source for PLL: disabled
- External Clock (Hz): 1 <= 3000000 <= 24 × 10<sup>6</sup>
- Prescaler Enable: ☐
- Prescaler Division: 2X
- System Clock Out Enable: ☐

**Pin Grid View**

Package:	Pin No:	44	45	46	47	48	1	2	3	4	5	6	7	8	9	10	11	12	13	16	17	18	19	20	21	22	23	24	25	26	27	30	31	32	33	34	35	36	37
CLKCTRL	CLKOUT																																						
	GPIO																																						
Pins	GPIO																																						
	GPIO																																						

# Ex1 的 MCC Project Setting

The screenshot displays the Microchip MCC v5.3.7 interface. The top menu bar includes options like Project Resources, Files, Services, Class, and Builder. The left sidebar shows the Project Resources tree with categories like Timer, System, and Device Resources. The central workspace is titled 'Show system firmware connections' and contains a grid with components: System Firmware, DELAY, and a Microchip AVR128A003 chip. The right sidebar shows the Pins table.

Location	Pin Name	Module	Function	Direction	Custom Name	Start High	Inverted
6	PB2	Pins	GPIO	output	LED1	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	PB3	Pins	GPIO	output	LED2	<input type="checkbox"/>	<input type="checkbox"/>
8	PB4	Pins	GPIO	output	LED3	<input type="checkbox"/>	<input type="checkbox"/>
9	PB5	Pins	GPIO	output	LED4	<input type="checkbox"/>	<input type="checkbox"/>

# Delay Driver 的使用說明

## ✓ 3.1.2 How to Use the Delay Driver

To use the delay drivers, include the [delay.h](#) file wherever the delay function is needed and call the `DELAY_milliseconds(time)` or `DELAY_microseconds(time)` function, where *time* is the number of milliseconds or microseconds to delay.

### ✓ 3.1.2.1 Use Cases

The Delay driver is an interface for abstracting the built-in delay macros.

#### ✓ 3.1.2.1.1 Use Case 1: Delay for 1s before Toggling an LED

Use the `DELAY_milliseconds` function defined by the Delay driver:

- This requires 1000 ms to get a 1s delay:

```
int main(void)
{
    SYSTEM_Initialize();

    while(1){
        DELAY_milliseconds(1000); //Delay for 1s
        LED_Toggle();
    }
}
```



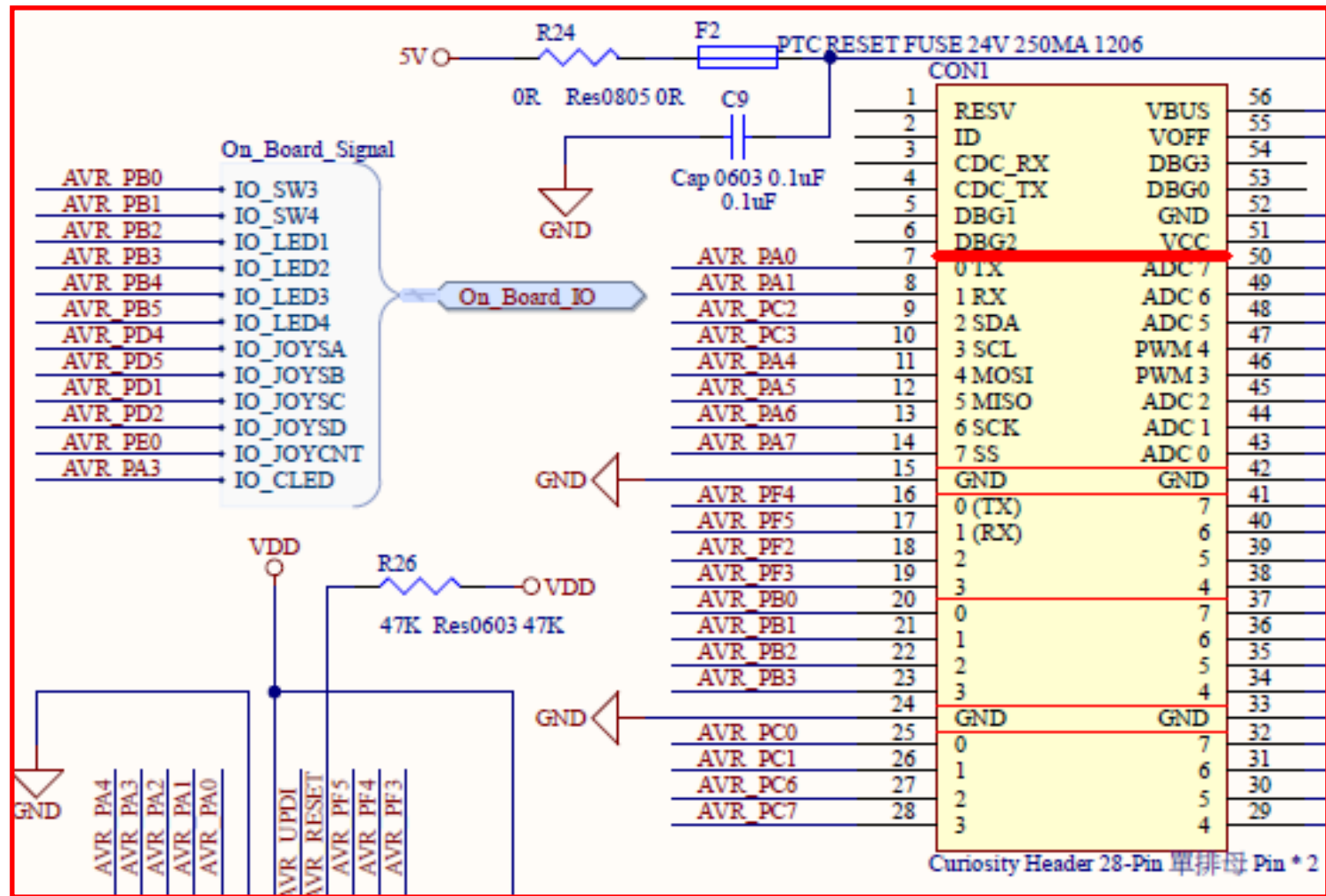
# Ex2 : AVR128DA48 RTC & Interrupt

---

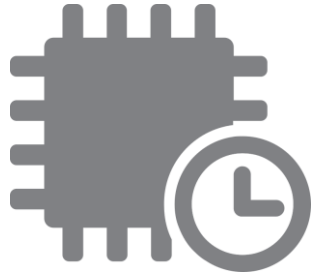
Use : MPLAB XIDE V6.10

# 練習二 Project Guide

- Project Name :  
Lab\_AVR128DA48
- Project Folder :
  - RTC\AVR101
- 使用的資源
  - AVR128DA48 控制 LED1 ~LED4
    - LED1 : AVR\_PB2
    - LED2 : AVR\_PB3
    - LED3 : AVR\_PB4
    - LED4 : AVR\_PB5
  - Melody 提供的 RTC Driver
- Ex2 要達成的功能
  - 完成一個每 100 ms 位移一次的跑馬燈
  - 位移的方向由 LED1 住次向 LED4，並且在盡頭轉換方向



# Timer/Counters



- **16-bit Timer/Counter A (TCA)**

- Flexible 16-bit PWM provides accurate program execution timing.
- 3 PWM channels

- **16-bit Timer/Counter B (TCB)**

- Include frequency and waveform generation, and input capture on Event with time and frequency measurement of digital signals.
- 1 PWM channel

- **TCA Application Example**

- LED RGB lighting
- 3-phase BLDC
- With external motor driver e.g. MTD6501

- **TCB Application Example**

- Capture signal via event (CIP)
- Hardware monitoring for missed periodic events and fault detection (stalls, stops, etc.)
- Safety limit in closed loop control applications
- Power supplies
- Motor driver

# Timer/Counters

- **12-bit Timer/Counter D (TCD)**

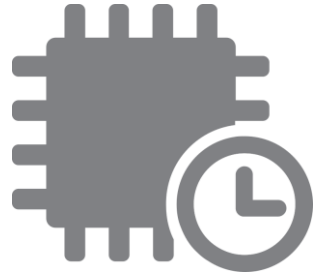
- High performance waveform controller consisting of an asynchronous counter, a prescaler, compare logic, capture logic, and control logic.
- Designed for control applications.
- 2 PWM channels

- **16-bit Real Time Counter (RTC)**

- Selectable clock source
- Periodic interrupt and event generation

- **TCD Application Example**

- H-bridge
- LED
- Power converters
- Motor control



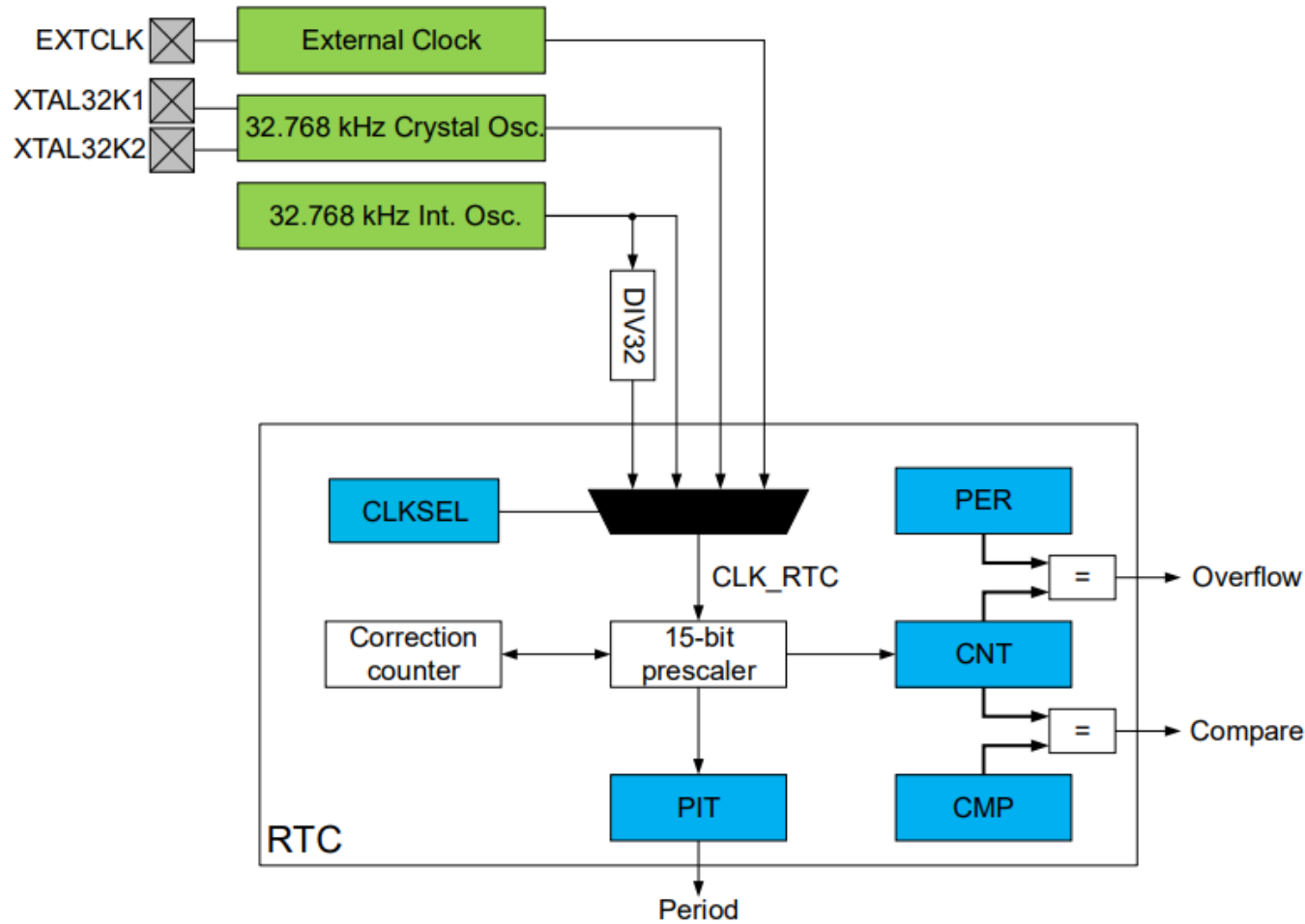
- **RTC Application Example**

- Creates a periodic Event or Interrupt to measure sensors





# RTC Module



## • RTC 重要諸元

- 16-bit Resolution
- Selectable Clock Sources
- Programmable 15-bit Clock Prescaling
- One Compare Register
- One Period Register
- Clear Timer on Period Overflow
- Optional Interrupt/Event on Overflow and Compare Match
- Periodic Interrupt and Event
- Crystal Error Correction

# RTC Module 會產生的 Events

## 可以用來產生中斷或作為 Event System 的觸發源

Generator Name		Description	Event Type	Clock Domain	Length of the Event
Module	Event				
RTC	OVF	Overflow	Pulse	CLK_RTC	One CLK_RTC period
	CMP	Compare Match			One CLK_RTC period
	PIT_DIV8192	Prescaled RTC clock divided by 8192	Level		Given by prescaled RTC clock divided by 8192
	PIT_DIV4096	Prescaled RTC clock divided by 4096			Given by prescaled RTC clock divided by 4096
	PIT_DIV2048	Prescaled RTC clock divided by 2048			Given by prescaled RTC clock divided by 2048
	PIT_DIV1024	Prescaled RTC clock divided by 1024			Given by prescaled RTC clock divided by 1024
	PIT_DIV512	Prescaled RTC clock divided by 512			Given by prescaled RTC clock divided by 512
	PIT_DIV256	Prescaled RTC clock divided by 256			Given by prescaled RTC clock divided by 256
	PIT_DIV128	Prescaled RTC clock divided by 128			Given by prescaled RTC clock divided by 128
	PIT_DIV64	Prescaled RTC clock divided by 64			Given by prescaled RTC clock divided by 64

# 中斷 : Interrupt

- 中斷是周邊用來反映其已達預先設置的狀態條件，向 **CPU** 要求服務的一種機制
- 每個周邊可能引發一種以上的中斷，例如 **RTC** 可以引發
  - Compare, Overflow, PIT
- 每個中斷都有獨立的中斷向量 (**Interrupt Vector**)
  - 指向服務該項中斷的“中斷服務程式”的起點
- 當然，只有被致能的中斷會引發 **Interrupt Request**
- **AVR** 的中斷特點：
  - 快速而且可預期的中斷反應時間

# 中斷向量表可以在 Data Sheet 中找到

Vector Number	Program Address (word)	Peripheral Source	Description	28-Pin	32-Pin	48-Pin	64-Pin
0	0x00	RESET		X	X	X	X
1	0x02	NMI	Non-Maskable Interrupt available for CRCSCAN	X	X	X	X
2	0x04	BOD_VLM	Voltage Level Monitor Interrupt	X	X	X	X
3	0x06	RTC_CNT	Real-Time Counter Overflow or Compare Match Interrupt	X	X	X	X
4	0x08	RTC_PIT	Real-Time Counter Periodic Interrupt	X	X	X	X
5	0x0A	CCL_CCL	Configurable Custom Logic Interrupt	X	X	X	X
6	0x0C	PORTA_PORT	PORTA External interrupt	X	X	X	X
7	0x0E	TCA0_OVF TCA0_LUNF	Normal: Timer/Counter Type A Overflow Interrupt Split: Timer/Counter Type A Low Underflow Interrupt	X	X	X	X
8	0x10	TCA0_HUNF	Normal: Unused Split: Timer/Counter Type A High Underflow Interrupt	X	X	X	X
9	0x12	TCA0_CMP0 TCA0_LCMP0	Normal: Timer/Counter Type A Compare 0 Interrupt Split: Timer/Counter Type A Low Compare 0 Interrupt	X	X	X	X
10	0x14	TCA0_CMP1 TCA0_LCMP1	Normal: Timer/Counter Type A Compare 1 Interrupt Split: Timer/Counter Type A Low Compare 1 Interrupt	X	X	X	X



# 中斷的優先權

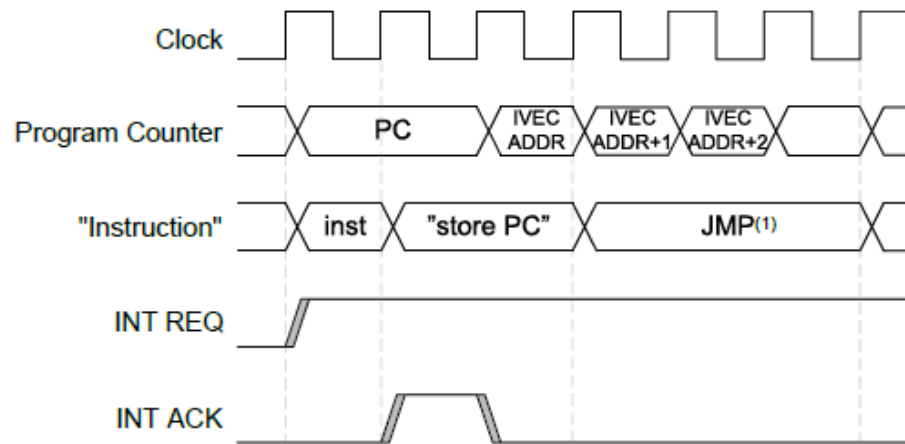
- 分為 : **Non-Maskable Interrupt** 、 **Level 1 & Level 2**
- **Non-Maskable interrupt** 來源可能在每個 **AVR MCU** 不相同
  - AVR128DA48 的 Source 是 CRCSCAN
- **Level 1** 的中斷只有一個
  - 設定方法為將中斷向量寫入 CPUINT.LVL1VEC 暫存器
- 其他為 **Level 0** 的中斷源，以中斷向量的前後來決定衝突時的先後順序

Priority	Level	Source
Highest	Non-Maskable Interrupt	Device-dependent and statically assigned
...	Level 1 (high priority)	One vector is optionally user selectable as level 1
Lowest	Level 0 (normal priority)	The remaining interrupt vectors

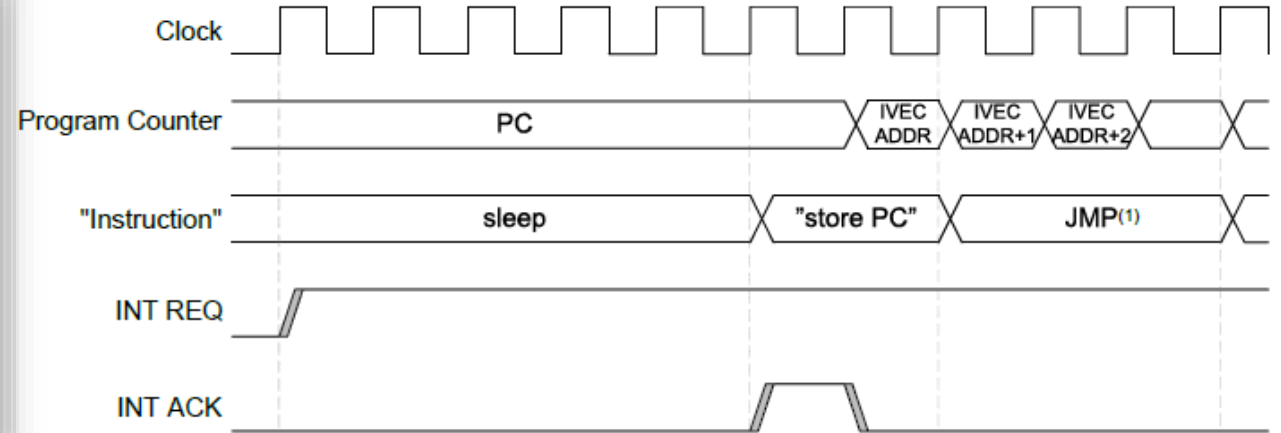
# AVR<sup>®</sup> MCU 快速而且可預期的中斷反應時間

	Flash Size > 8 KB	Flash Size ≤ 8 KB
Finish ongoing instruction	One cycle	One cycle
Store PC to stack	Two cycles	Two cycles
Jump to interrupt handler	Three cycles (jmp)	Two cycles (rjmp)

Interrupt Execution of Single-Cycle Instruction



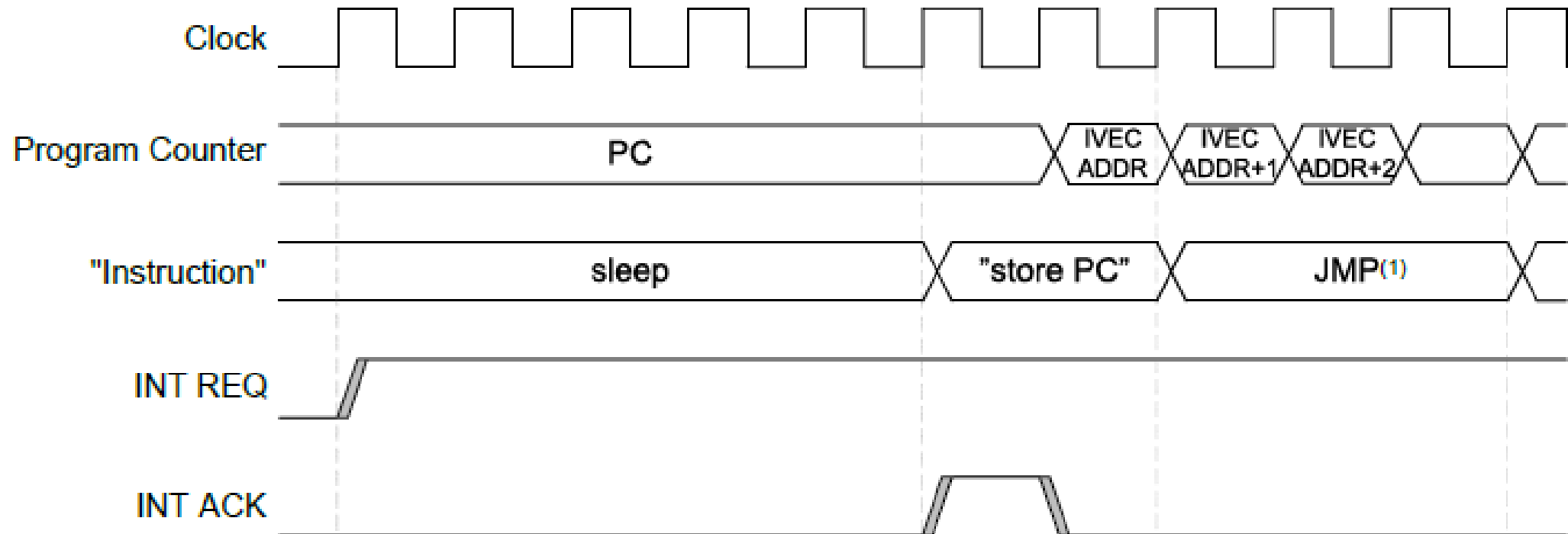
Interrupt Execution From Sleep



# CPU 在 Sleep 下的中斷喚醒

- 在 Sleep Mode 下仍然可以工作的周邊可以喚醒 CPU
- 因為 CPU 處於 Sleep，所以喚醒的時間會多 4 ~ 5 個 Clock

## . Interrupt Execution From Sleep



# MCC Melody 提供的中斷服務程式指定 API

```
void RTC_CMP_Callback(void){  
    LED_SetHigh();}
```

```
void RTC_OVF_Callback(void){  
    LED_SetLow();  
}
```

```
int main(void)  
{  
    SYSTEM_Initialize();  
  
    RTC_SetCMPISrCallback(RTC_CMP_Callback);  
    RTC_SetOVFISrCallback(RTC_OVF_Callback);  
  
    while(1)  
    {  
    }  
}
```



# Lets' go Ex2 by following the procedures in Video 😊

<https://youtu.be/1tqvoZbnKqM>

自動儲存 (關閉) 44 AVR101\_AVR128DA48... • 已儲存到此電腦 搜尋 Calvin Ho - A90045

檔案 常用 插入 繪圖 設計 轉場 動畫 投影片放映 錄製 校閱 檢視 增益集 說明

剪貼簿 投影片 字型 段落 繪圖 編輯 語言 設計工具

## 練習二 Project Guide

- Project Name :  
Lab\_AVR128DA48
- Project Folder :
  - MCHP\_ProjDev\_2023\AVR128DA48
- 使用的資源
  - AVR128DA48 控制 LED1 ~LED4
    - LED1: AVR\_PB2
    - LED2: AVR\_PB3
    - LED3: AVR\_PB4
    - LED4: AVR\_PB5
  - Melody 提供的 RTC Driver
- Ex2 要達成的功能
  - 完成一個每 100 ms 位移一次的跑馬燈
  - 位移的方向由 LED1 住次向 LED4 , 並且在盡頭轉換方向

Pin	Signal	Pin	Signal
1	RESV	56	VBUS
2	ID	55	VOFF
3	CDC_RX	54	DBG3
4	CDC_TX	53	DBG0
5	DBG1	52	GND
6	DBG2	51	VCC
7	0 TX	50	ADC 7
8	1 RX	49	ADC 6
9	2 SDA	48	ADC 5
10	3 SCL	47	PWM 4
11	4 MOSI	46	PWM 3
12	5 MISO	45	ADC 2
13	6 SCK	44	ADC 1
14	7 SS	43	ADC 0
15	GND	42	GND
16	0 (TX)	41	GND
17	1 (RX)	40	GND
18	2	39	GND
19	3	38	GND
20	4	37	GND
21	5	36	GND
22	6	35	GND
23	7	34	GND
24	8	33	GND
25	9	32	GND
26	10	31	GND
27	11	30	GND
28	12	29	GND

Curiosity Header 28-Pin 單排每 Pin \* 2

Microchip Proprietary and Confidential

投影片 78 / 146 繁體中文 (台灣) 協助工具: 調查 復原 備忘稿 85%

在 這裡輸入文字來搜 35°C 下午 02:52 2023/9/19

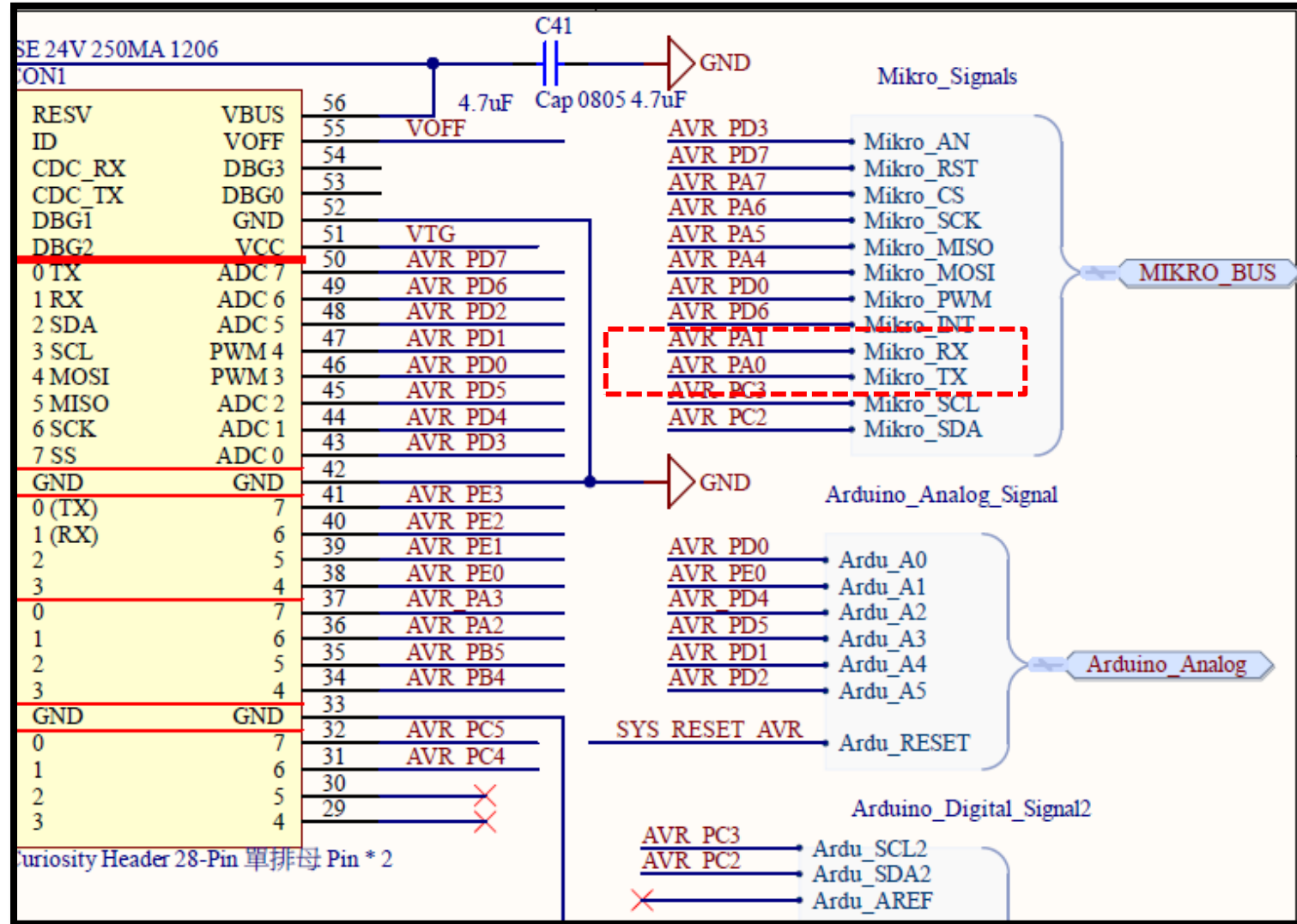
# Ex3 : AVR128DA48 USART

---

Use : MPLAB XIDE V6.10

# 練習三 Project Guide

- **Project Name :**  
**Lab\_AVR128DA48**
- **Project Folder :**
  - RTC\AVR101
- **使用的資源**
  - AVR128DA48 控制 LED1 ~LED4
    - LED1 ~ LED4
  - Melody 提供的 **RTC Driver**
  - Melody 提供的 **UART Driver**
    - **115200 bps , UART @ UART0**
- **Ex3 要達成的功能**
  - 完成一個每 **200 ms** 位移一次的跑馬燈以及 **Hello Word** 的列印
    - 位移的方向由 LED1 住次向 LED4，並且在盡頭轉換方向



# 使用 Data Visualizer 來觀察 UART 的輸出

- **DV** 是 MPLAB X IDE 安裝時會一併安裝的 Plug-In

The screenshot displays the MPLAB X IDE interface with the Data Visualizer (DV) plug-in active. The top toolbar includes various icons for file operations, execution, and debugging. The workspace shows the project files, including 'main.c'. The DV interface is divided into several sections:

- Serial Ports:** Shows 'COM8' as the selected serial port, with a 'Serial' streamer and a 'CAPTURING' status indicator.
- Time Plot:** A graph showing data values over time. The Y-axis ranges from 0 to 200, and the X-axis shows time markers at 0s, 5s, and 10s. A single data point is visible at 0s with a value of 'n/a'.
- Terminal:** Displays a log of messages: 'I am Calvin' followed by values 916 through 927.
- Configuration Panel (UART0):** Shows the settings for the selected UART device. Key parameters include:
  - Custom Name: UART0
  - Requested Baudrate: 115200
  - Calculated Baudrate: 115246
  - Baud Rate Error (%): 0.04
  - Parity: None
  - Data Size: 8
  - Stop Bits: 1
  - Flow Control Mode: None
  - Redirect Printf to UART: Enabled (checked)
  - Interrupt Settings: Interrupt Driven (unchecked)
  - Dependency Selector: UART PLIB Selector (USART0)



# Serial Communication

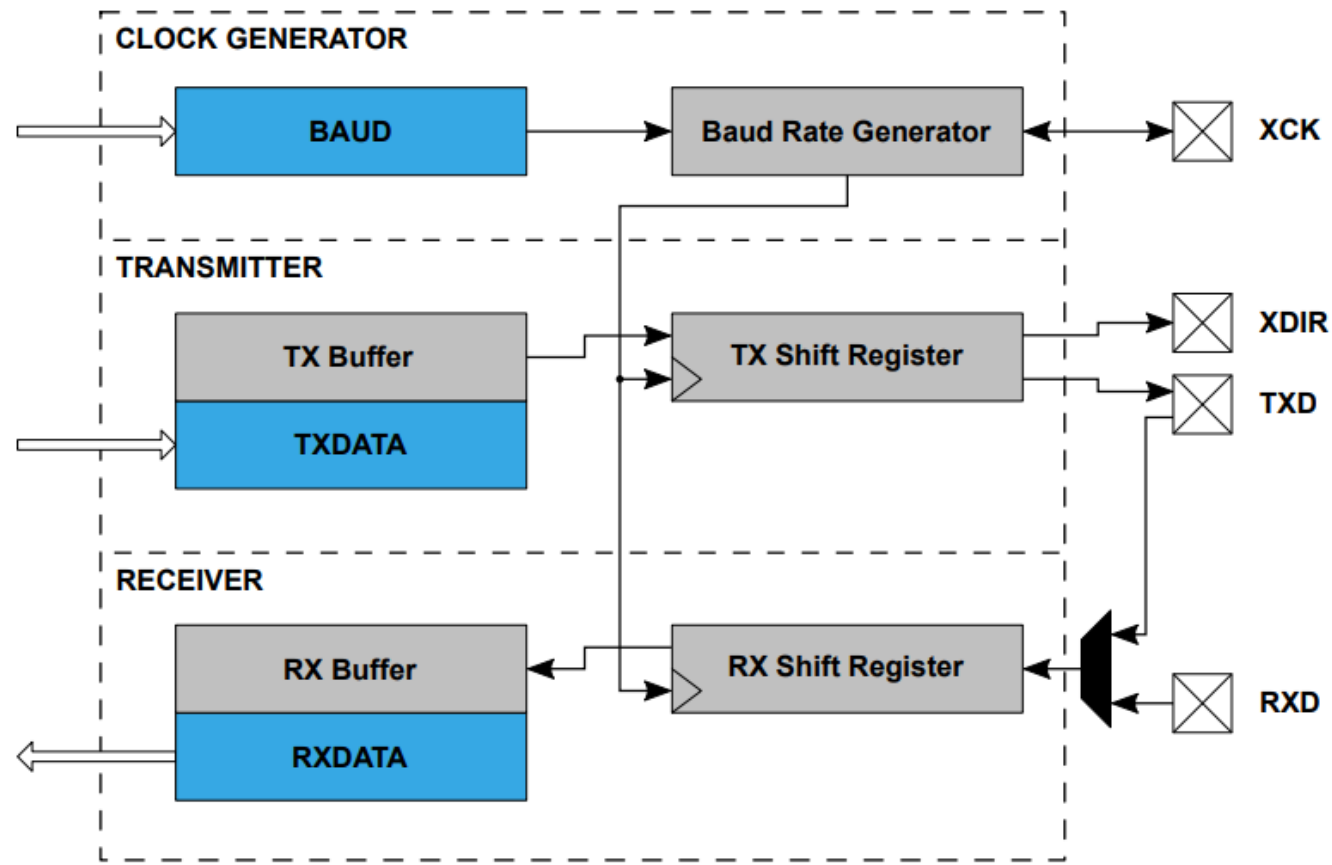
- **Up to six USARTs**
  - Operation modes: RS485, LIN client, host SPI and IrDA
  - Fractional baud rate generator, autobaud, and start-of-frame detection
- **Two SPI**
  - Host/client operation modes
- **Two I<sup>2</sup>C**
  - Simultaneous host/client operation and dual address match
  - Supported modes
    - Standard mode (100kHz)
    - Fast mode (400kHz)
    - Fast mode plus (1MHz)

- **Application Example**

- Board control
- Sensors



# AVR128DA48 USART 的方塊圖



Signal	Type	Description
XCK	Output/input	Clock for synchronous operation
XDIR	Output	Transmit enable for RS-485
TxD	Output/input	Transmitting line (and receiving line in One-Wire mode)
RxD	Input	Receiving line

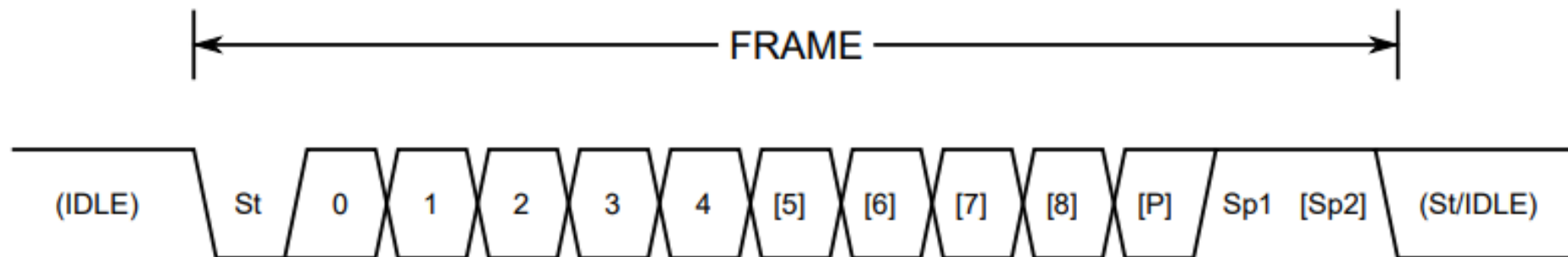
# AVR128DA48 USART 可以支援的 Data Frame

比較常用的傳輸模式為“非同步模式” Baud Rate 要兩邊一致

- 1 Start bit
- 5, 6, 7, 8, or 9 data bits
- No, even, or odd Parity bit
- 1 or 2 Stop bits

The figure below illustrates the possible combinations of frame formats. Bits inside brackets are optional.

Figure 25-2. Frame Formats



**St** Start bit, always low

**(n)** Data bits (0 to 8)

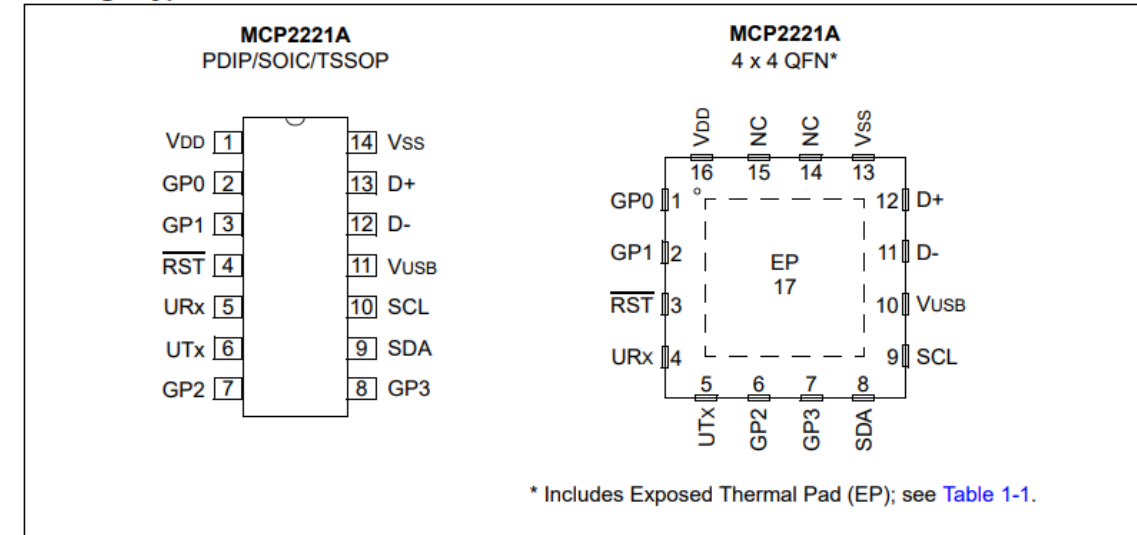
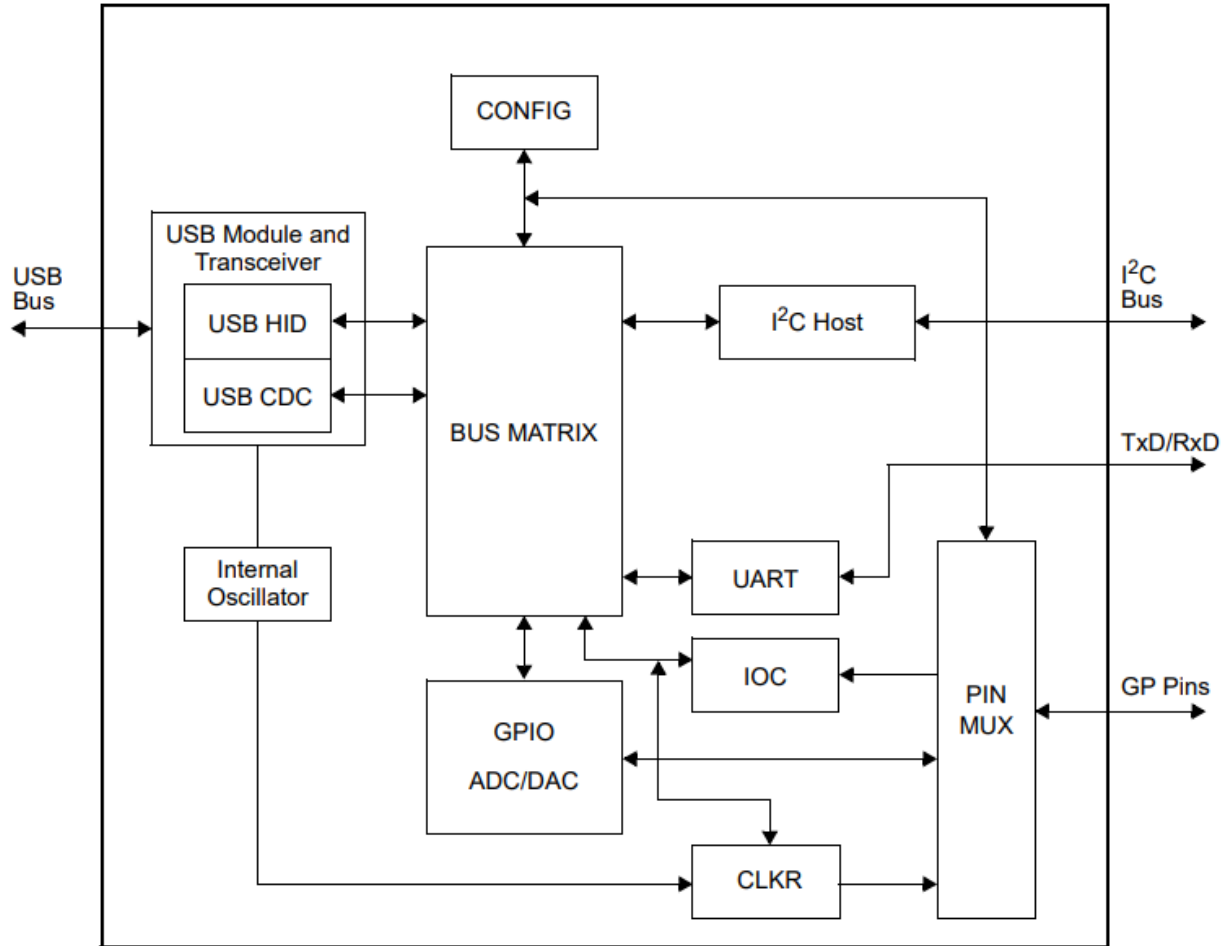
**P** Parity bit, may be odd or even

**Sp** Stop bit, always high

**IDLE** No transfer on the communication line (RxD or TxD). The Idle state is always high.

# Microchip MCP2221A

一個 USB-to-UART/I<sup>2</sup>C 的轉器，讓更多的應用能與 USB 相連

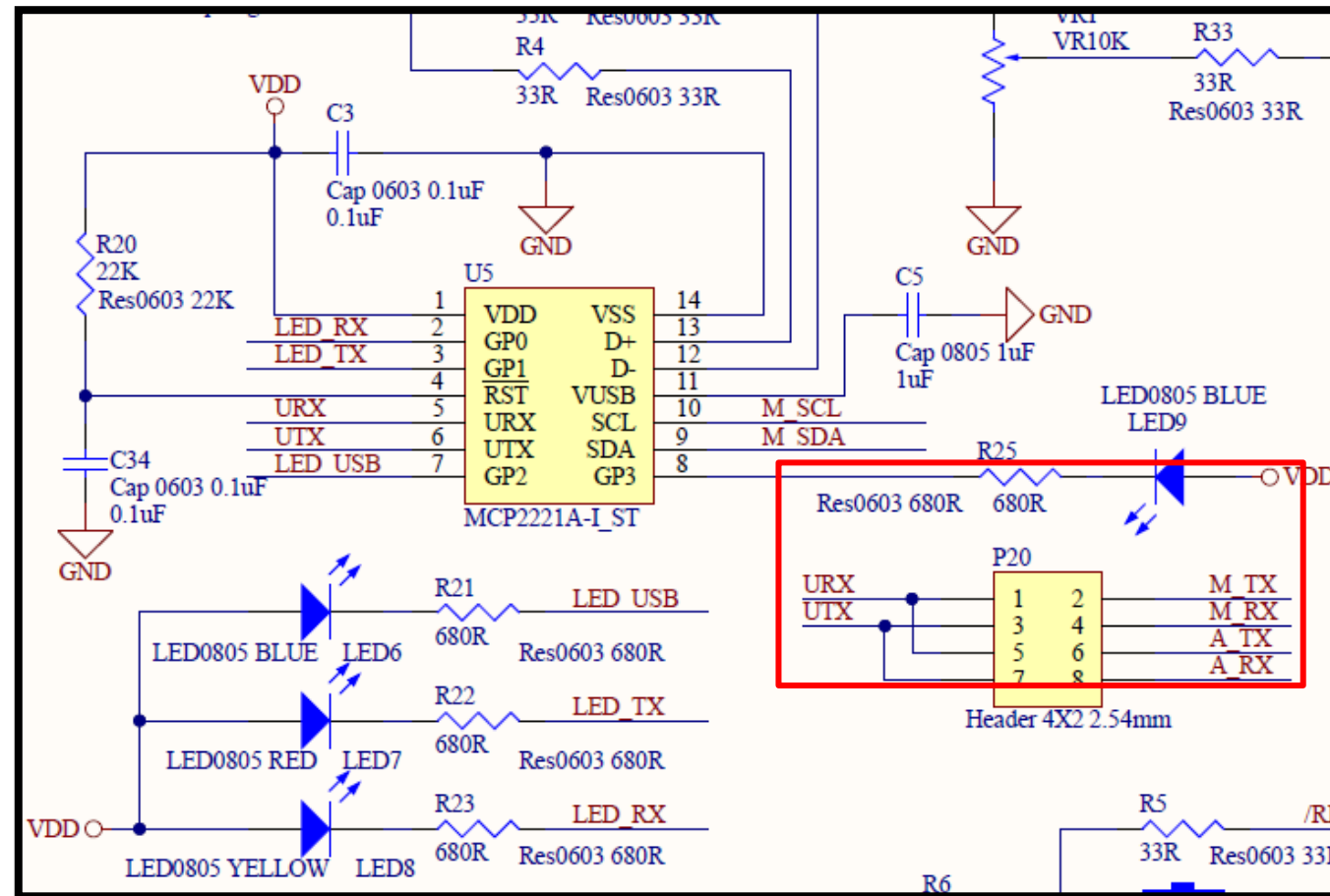




# MCP2221A 支援的 UART 通信速率

透過 P20，MCP2221A 可以支援兩組 UART 信號的收送

Desired Rate	Actual rate	% Error
300	300	0.00%
1200	1200	0.00%
2400	2400	0.00%
4800	4800	0.00%
9600	9600	0.00%
19200	19200	0.00%
38400	38339	0.16%
57600	57692	0.16%
115200	115385	0.16%
230400	230769	0.16%
460800	461538	0.16%



# Lets' go Ex3 by following the procedures in Video ☺

<https://youtu.be/dFvL6fvSdgc>

自動儲存 (關閉) AVR101\_AVR128DA48... • 已儲存到此電腦 搜尋 Calvin Ho - A90045

檔案 常用 插入 繪圖 設計 轉場 動畫 投影片放映 錄製 校閱 檢視 增益集 說明

剪貼簿 貼上 新 重複使 用投影片 版面配置 重設 章節

字型 段落 繪圖 編輯 設計工具

## 練習三 Project Guide

- Project Name : Lab\_AVR128DA48
- Project Folder : MCHP\_ProjDev\_2023\AVR128DA48
- 使用的資源
  - AVR128DA48 控制 LED1 ~ LED4
    - LED1 ~ LED4
  - Melody 提供的 **RTC Driver**
  - Melody 提供的 **UART Driver**
    - 115200 bps, UART @ UART0
- Ex3 要達成的功能
  - 完成一個每 **200 ms** 位移一次的跑馬燈以及 **Hello Word** 的列印
    - 位移的方向由 LED1 住次向 LED4, 並且在盡頭轉換方向

SE 24V 250MA 1206  
ON1

RESV VBUS 56  
ID VOFF 55  
CDC\_RX DBG3 54  
CDC\_TX DBG0 53  
DBG1 GND 52  
DBG2 VCC 51  
0 TX ADC 7 50  
1 RX ADC 6 49  
2 SDA ADC 5 48  
3 SCL PWM 4 47  
4 MOSI PWM 3 46  
5 MISO ADC 2 45  
6 SCK ADC 1 44  
7 SS ADC 0 43  
GND GND 42  
0 (TX) 7 41  
1 (RX) 6 40  
2 5 39  
3 4 38  
0 7 37  
1 6 36  
2 5 35  
3 4 34  
GND GND 33  
0 7 32  
1 6 31  
2 5 30  
3 4 29

4.7uF 4.7uF  
Cap 0805 4.7uF

AVR PD3  
AVR PD7  
AVR PA7  
AVR PA6  
AVR PA5  
AVR PA4  
AVR PD0  
AVR PD6  
AVR PA1  
AVR PA0  
AVR PC3  
AVR PC2

Mikro\_Signals  
Mikro\_AN  
Mikro\_RST  
Mikro\_CS  
Mikro\_SCK  
Mikro\_MISO  
Mikro\_MOSI  
Mikro\_PWM  
Mikro\_INT  
Mikro\_RX  
Mikro\_TX  
Mikro\_SCL  
Mikro\_SDA

MIKRO\_BUS

AVR PE3  
AVR PE2  
AVR PE1  
AVR PA3  
AVR PA2  
AVR PB5  
AVR PB4

Arduino\_Analog\_Signal  
Ardu\_A0  
Ardu\_A1  
Ardu\_A2  
Ardu\_A3  
Ardu\_A4  
Ardu\_A5

Arduino\_Analog

AVR PD0  
AVR PE0  
AVR PD4  
AVR PD5  
AVR PD1  
AVR PD2

Ardu\_RESET

Arduino\_Digital\_Signal2  
Ardu\_SCL2  
Ardu\_SDA2  
Ardu\_AREF

SYS RESET AVR

AVR PC3  
AVR PC2

Ardu\_SCL2  
Ardu\_SDA2  
Ardu\_AREF

Curiosity Header 28-Pin 單排母 Pin \* 2

MICROCHIP

Microchip Proprietary and Confidential

投影片 90 / 152 繁體中文 (台灣) 協助工具: 調查 復原 備忘稿 85%

在這裡輸入文字來搜 35°C 下午 03:50 2023/9/19

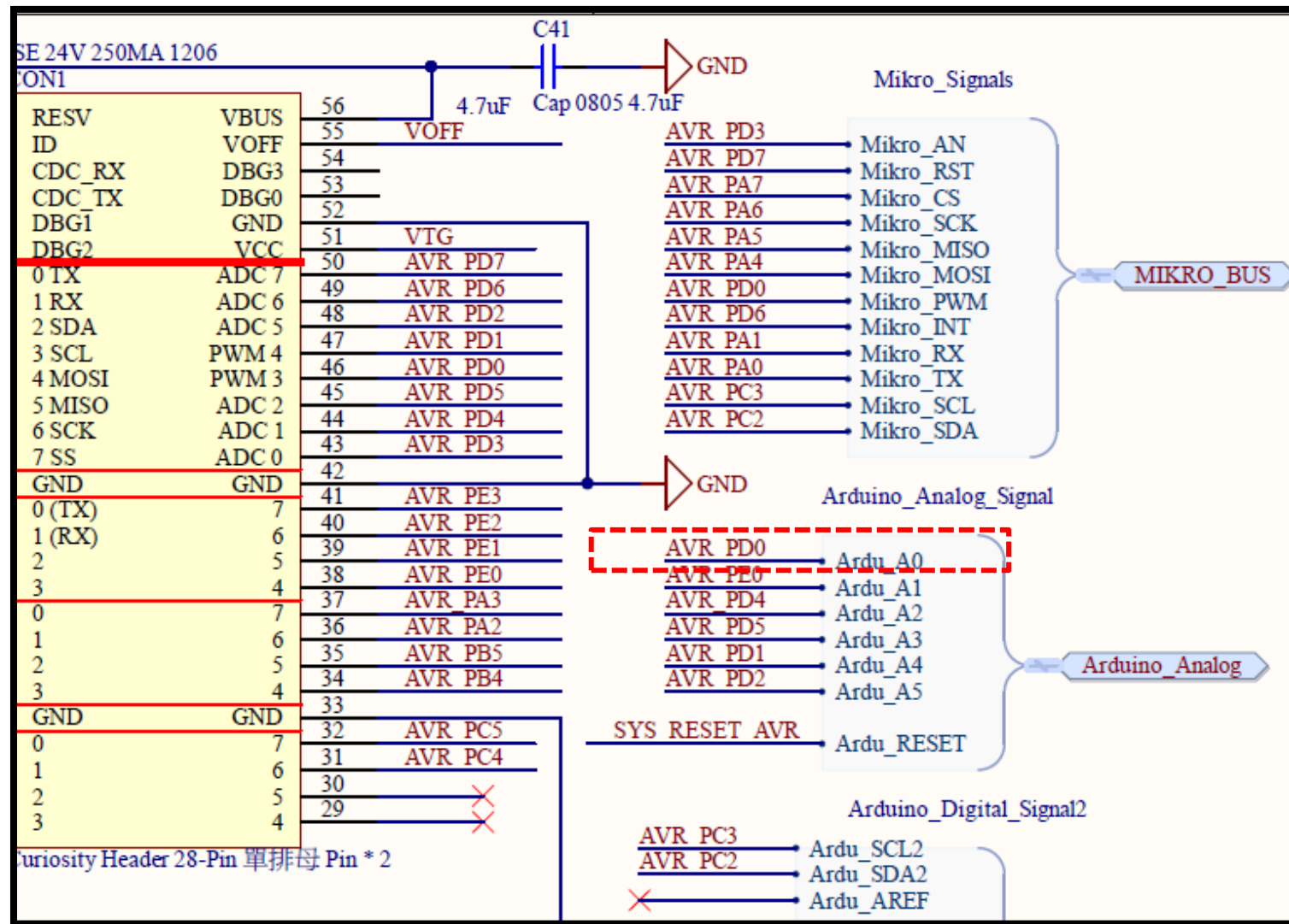
# Ex4 : AVR128DA48 ADC

---

Use : MPLAB XIDE V6.10

# 練習四 Project Guide

- **Project Name :**  
**Lab\_AVR128DA48**
- **Project Folder :**
  - MCHP\_ProjDev\_2023\AVR128DA48
- **使用的資源**
  - AVR128DA48 控制 LED1 ~LED4 , VR1
    - LED1 ~ LED4
    - VR1 @ PD0
  - Melody 提供的 **RTC Driver**
  - Melody 提供的 **UART Driver**
    - 115200 bps , UART @ UART0
  - Melody 提供的 **ADC Driver**
- **Ex4 要達成的功能**
  - 完成一個每 **200 ms** 位移一次的跑馬燈以及 **ADC 結果** 的列印
    - 位移的方向由 LED1 住次向 LED4 , 並且在盡頭轉換方向





# 練習四在 DV 的輸出實況

MPLAB Data Visualizer interface showing the output of a program. The interface includes a workspace with files like `main.c` and `Builder`, and a toolbar with various icons.

The **Serial Ports** panel shows **COM8** is **CAPTURING**.

The **Time Plot** panel displays a graph with a marker at 0s and data values. The Y-axis ranges from 0 to 200, and the X-axis ranges from 0s to 10s. A legend indicates the marker is at 0s and the data values are n/a.

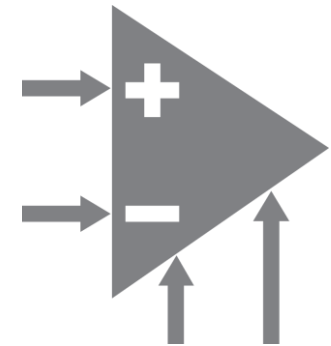
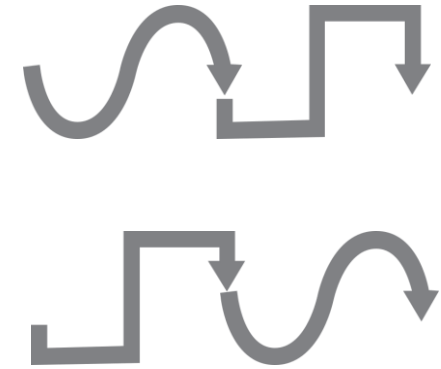
The **Terminal** panel shows the output of the program, displaying multiple `ADC Result` values (1727 and 1726) and a `Line input` field.

The **Pin Configuration** table lists the pins and their configurations:

Pin	Module	Function	Direction	Custom Name	Start High	Inverted I/O	Pull-up	Input/Sense Configuration [ISC]
PA1	USART0	RXD	input	IO_PA1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...
PA0	USART0	TXD	output	IO_PA0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...
PD0	ADC0	AINx	input	IO_PD0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Digital Input...
PB2	Pins	GPIO	output	LED1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...
PB3	Pins	GPIO	output	LED2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...
PB4	Pins	GPIO	output	LED3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...
PB5	Pins	GPIO	output	LED4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interrupt dis...

# Intelligent Analog

- **One 12-bit Analog to Digital Converter (ADC)**
  - Single-ended and differential mode
  - 130 ksp/s
- **One 10-bit Digital to Analog Converter (DAC)**
  - Accurate set up for current protection.
- **Internal voltage references**
  - Selectable levels: 1.024V, 2.048V, 4.096V and 2.500V and external reference option
- **Three Analog Comparators (ACs)**
  - Use for overcurrent detection for motor control, lighting and gas blowers
  - Individual DAC input voltage reference
- **Three Zero Cross Detectors (ZCD)**
  - Detect zero crossings on high-voltage alternating signals
  - Drive outputs such as LED indicator, relay, control gate or switching purposes in power electronics.

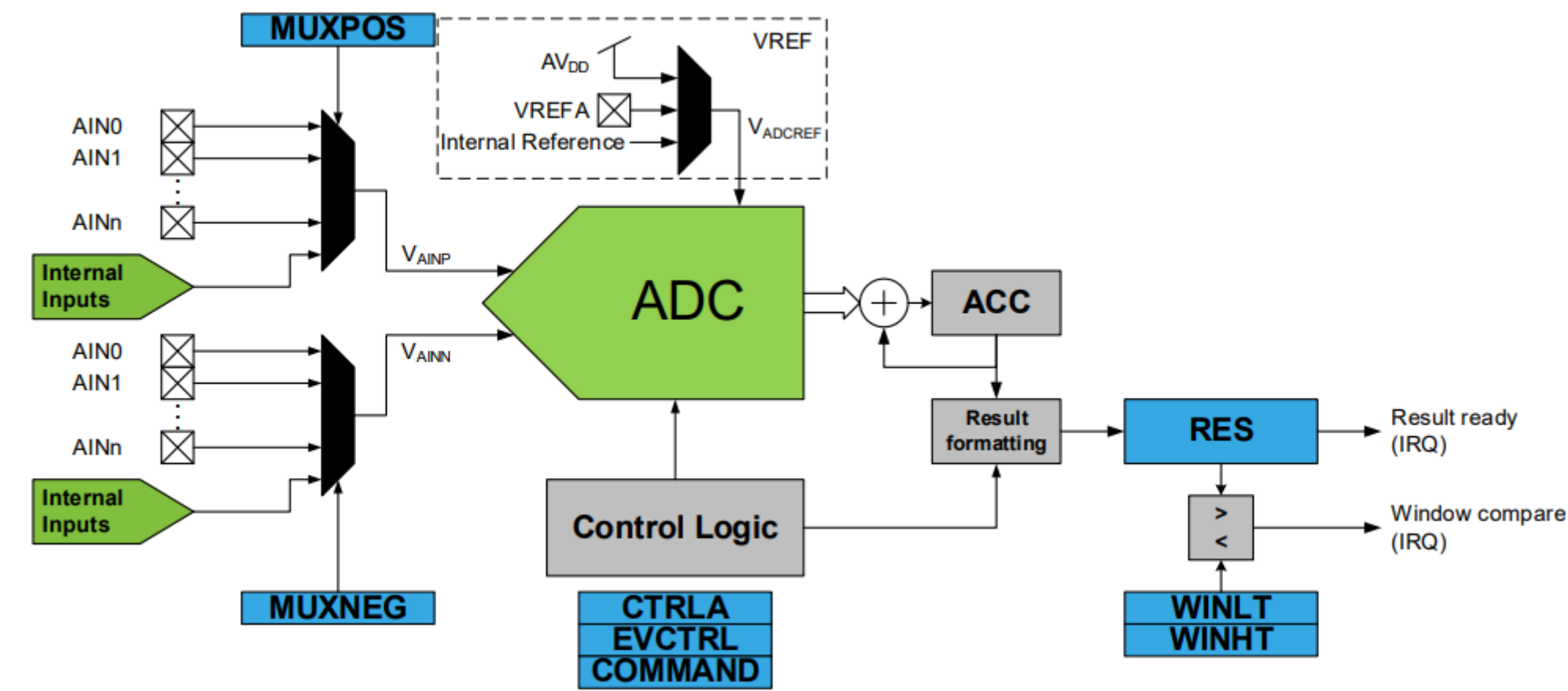


# AVR128DA48 ADC 模組的功能及特性

一般而言，物理量為類比形式，必須轉換為數位之後進行處理

- **12-Bit Resolution - Successive Approximation Register (SAR) ADC**
- **Up to 130 ksps at 12-Bit Resolution**
- **Differential and Single-Ended Conversion**
- **Up to 22 Inputs**
- **Rail-to-Rail Input Voltage Range**
- **Free-Running and Single Conversion**
- **Accumulation of Up to 128 Samples per Conversion**
- **Multiple Voltage Reference Options**
- **Temperature Sensor Input Channel**
- **Programmable Input Sampling Duration**
- **Configurable Threshold and Window Comparator**
- **Event Triggered Conversion • Interrupt and Event on Conversion Complete**

# AVR128DA48 ADC 的方塊圖



Pin Name	Type	Description
AIN[n:0]	Analog input	Analog input to be converted
VREFA	Analog input	External voltage reference pin



# 如果沒有 MCC，User 使用 ADC 前必須完成的工作

1. Configure the ADC voltage reference in the Voltage Reference (VREF) peripheral.
2. Optional: Select between Single-Ended or Differential mode by writing to the Conversion Mode (CONVMODE) bit in the Control A (ADCn.CTRLA) register.
3. Configure the resolution by writing to the Resolution Selection (RESSEL) bit field in the ADCn.CTRLA register.
4. Optional: Configure to left adjust by writing a '1' to the Left Adjust Result (LEFTADJ) bit in the ADCn.CTRLA register.
5. Optional: Select the Free-Running mode by writing a '1' to the Free-Running (FREERUN) bit in the ADCn.CTRLA register.
6. Optional: Configure the number of samples to be accumulated per conversion by writing to the Sample Accumulation Number Select (SAMPNUM) bit field in the Control B (ADCn.CTRLB) register.
7. Configure the ADC clock (CLK\_ADC) by writing to the Prescaler (PRESC) bit field in the Control C (ADCn.CTRLC) register.
8. Select the positive ADC input by writing to the MUXPOS bit field in the ADCn.MUXPOS register.
9. Optional: Select the negative ADC input by writing to the MUXNEG bit field in the ADCn.MUXNEG register.
10. Optional: Enable Start Event input by writing a '1' to the Start Event Input (STARTEI) bit in the Event Control (ADCn.EVCTRL) register, and configure the Event System accordingly.
11. Enable the ADC by writing a '1' to the ADC Enable (ENABLE) bit in the ADCn.CTRLA register.

# ADC0\_GetConversion 的內容

```
adc_result_t ADC0_GetConversion( adc_0_channel_t channel)
{
    adc_result_t res;

    ADC0_StartConversion( channel);
    while ( !ADC0_IsConversionDone());
    res      = ADC0_GetConversionResult();
    ADC0.INTFLAGS = ADC_RESRDY_bm;
    return res;
}
```

# Lets' go Ex4 by following the procedures in Video ☺

<https://youtu.be/FrHPZgRIXjs>

自動儲存 (關閉) AVR101\_AVR128DA48... 已儲存到此電腦 搜尋 Calvin Ho - A90045

檔案 常用 插入 繪圖 設計 轉場 動畫 投影片放映 錄製 校閱 檢視 增益集 說明

貼上 剪貼簿 新 重複使用 重設 版面配置 投影片 用投影片 章節

字型 段落 繪圖 編輯 語音 設計工具

Ex4 : AVR128DA48 ADC

## 練習四 Project Guide

- Project Name : Lab\_AVR128DA48
- Project Folder :
  - MCHP\_ProjDev\_2023\AVR128DA48
- 使用的資源
  - AVR128DA48 控制 LED1 ~ LED4, VR1
    - LED1 ~ LED4
    - VR1 @ PD0
  - Melody 提供的 RTC Driver
  - Melody 提供的 UART Driver
    - 115200 bps, UART @ UART0
  - Melody 提供的 ADC Driver
- Ex4 要達成的功能
  - 完成一個每 200 ms 位移一次的跑馬燈以及 ADC 結果的列印
    - 位移的方向由 LED1 住次向 LED4, 並且在盡頭轉換方向

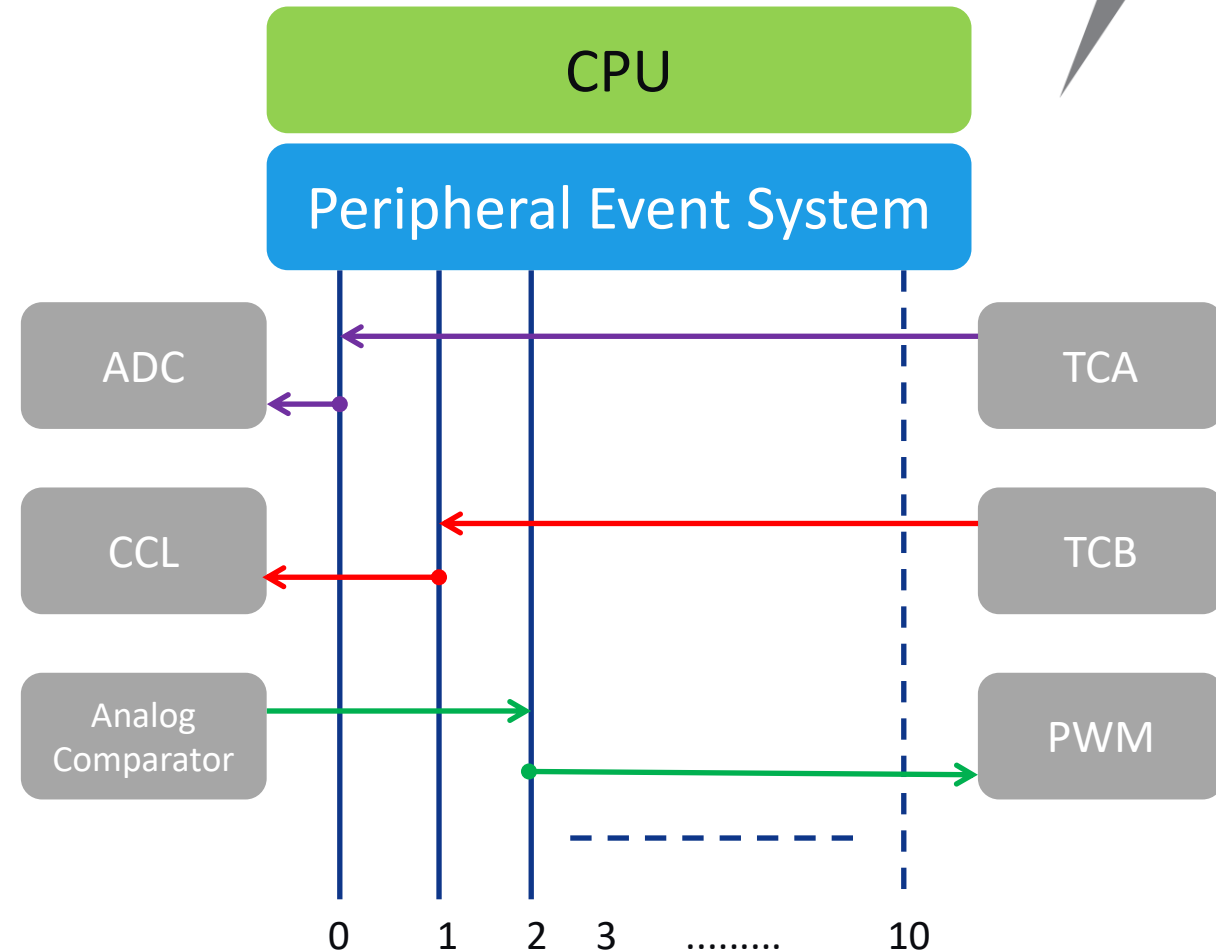
Pin	Function	Pin	Function
RESV	VBUS	56	VOFF
ID	VOFF	55	DBG3
CDC_RX	DBG3	53	DBG0
CDC_TX	DBG0	52	GND
DBG1	GND	51	VCC
DBG2	VCC	50	VTG
0 TX	ADC 7	49	AVR PD7
1 RX	ADC 6	48	AVR PD6
2 SDA	ADC 5	47	AVR PD2
3 SCL	PWM 3	46	AVR PD1
4 MOSI	PWM 3	45	AVR PD0
5 MISO	ADC 2	44	AVR PD5
6 SCK	ADC 1	43	AVR PD4
7 SS	ADC 0	42	AVR PD3
GND	GND	41	AVR PE3
0 (TX)	7	40	AVR PE2
1 (RX)	6	39	AVR PE1
2	5	38	AVR PE0
3	4	37	AVR PA3
0	7	36	AVR PA2
1	6	35	AVR PA1
2	5	34	AVR PA0
3	4	33	AVR PB4
GND	GND	32	AVR PB3
0	7	31	AVR PB2
1	6	30	AVR PB1
2	5	29	AVR PB0
3	4		

Microchip Proprietary and Confidential

112

# Event System

- **Core Independent Peripherals**
  - Executing task without CPU usage
  - Accurate and fast timing
  - Reduced power consumption
- **Latency-free Event handling**
  - Safe fault protection
  - Predictable reaction time
- **Inter-peripheral communication**
  - Up to 10 channels
  - Available in Active and Sleep modes
  - Events are never lost!





# 在 MPLAB Melody 對 EVSYS 的 Support

EVSYX x

Easy View Register Initialization

Software Settings

Custom Name Module

GENERATORS	CHANNELS	USERS
<div>CHANNEL0</div> <div>OFF</div> <div>UPDI_SYNC</div> <div>RTC_OVF</div> <div>RTC_CMP</div> <div>RTC_PIT_DIV8192</div> <div>RTC_PIT_DIV4096</div> <div>RTC_PIT_DIV2048</div> <div>RTC_PIT_DIV1024</div> <div>CCL_LUT0</div>	<div>CHANNEL0</div> <div>CHANNEL1</div> <div>CHANNEL2</div> <div>CHANNEL3</div> <div>CHANNEL4</div> <div>CHANNEL5</div> <div>CHANNEL6</div> <div>CHANNEL7</div> <div>CHANNEL8</div> <div>CHANNEL9</div>	<div>ADC0START</div> <div>CCLLUT0A</div> <div>CCLLUT0B</div> <div>CCLLUT1A</div> <div>CCLLUT1B</div> <div>CCLLUT2A</div> <div>CCLLUT2B</div> <div>CCLLUT3A</div> <div>CCLLUT3B</div> <div>CCLLUT4A</div>

Diagram illustrating the configuration of EVSYS modules in MPLAB Melody. The interface shows three main sections: GENERATORS, CHANNELS, and USERS. A blue line connects the **RTC\_PIT\_DIV8192** generator to the **CHANNEL0** channel. A black line connects the **CHANNEL0** channel to the **ADC0START** user.

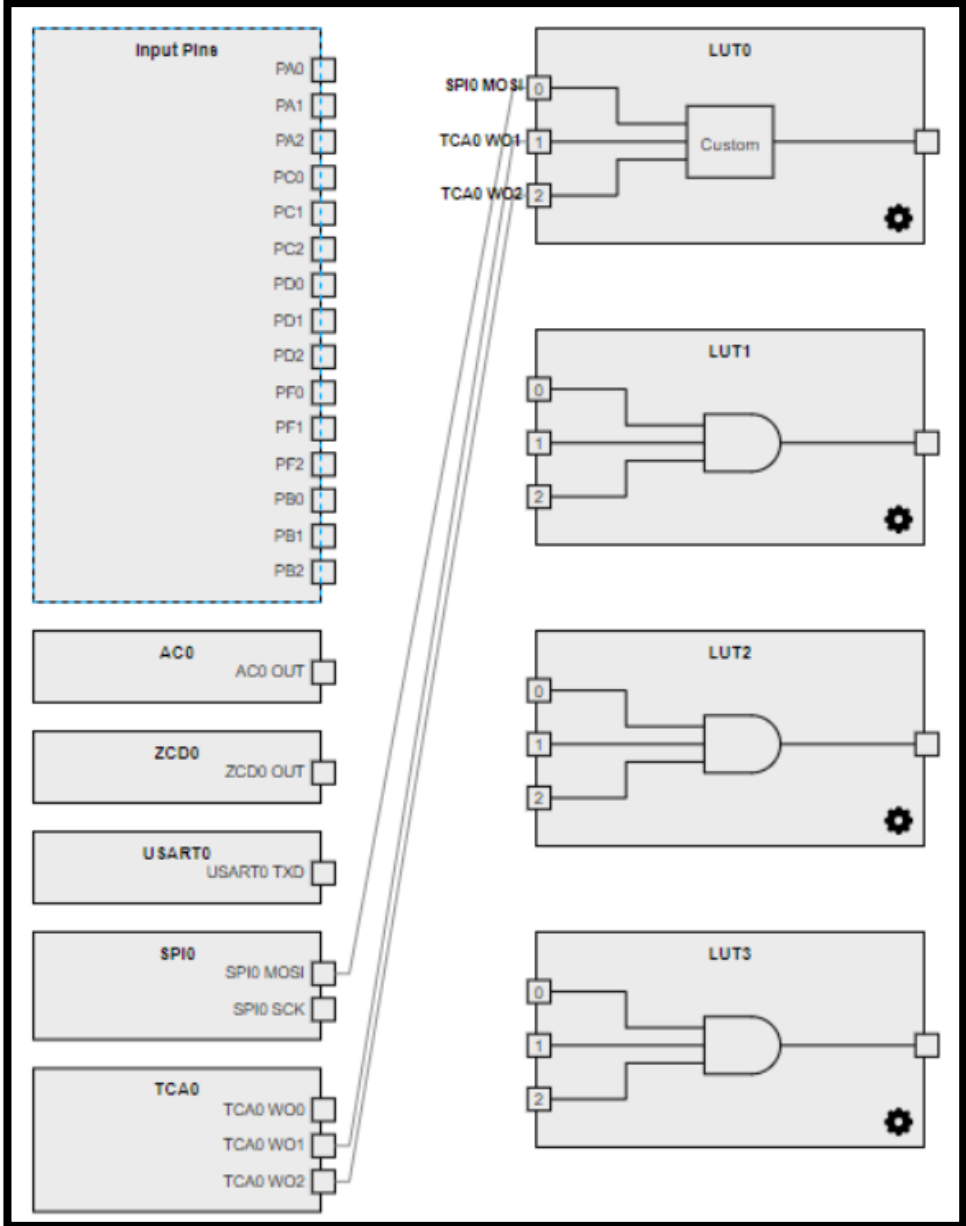
# Configure Custom Logic (CCL)



- Replace external logic and save execution power and BoM cost
- Configure performance critical tasks in hardware instead of software
- **Combinatorial or Sequential**
  - AND, NAND, OR, NOR, XOR, XNOR, NOT, MUX, RS Latch, D-Flip-Flop & D Latch
- **Functionality**
  - Two 2-input units
  - Connect output to external I/O pins or event system
  - Optional synchronizer
  - Filter or edge detector available on each output

Input			Output
A	B	C	Z
0	0	0	0   1
0	0	1	0   1
0	1	0	0   1
⋮			
1	1	1	0   1

# MCC Melody 可以對 CCL 做圖型化的互動設定



Clock Selection: CLKPER

Sequential Logic: DISABLE

Input 0 Source Selection: SPI0

Input 1 Source Selection: TCA0

Input 2 Source Selection: TCA0

Gate Type: Custom

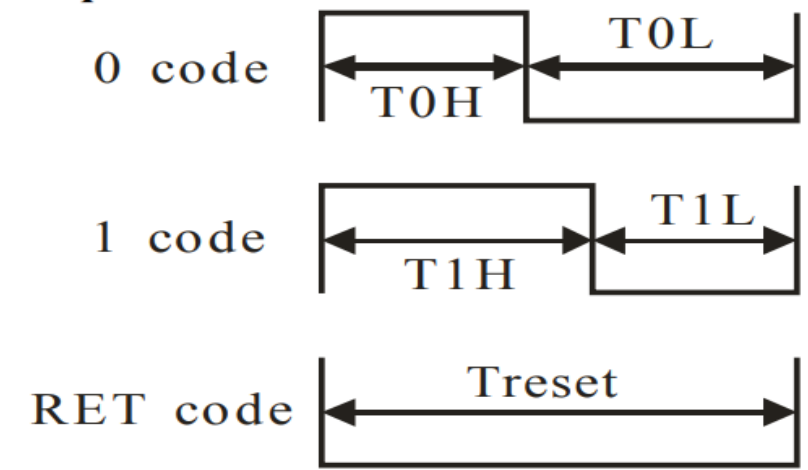
IN2	IN1	IN0	OUT
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

# CCL Module 的應用例 : WS2812

WS2812 : Timing Requirement 需要 800Khz 的控制訊號 (PWM)

Data transfer time( TH+TL=1.25μs±600ns)			
T0H	0 code ,high voltage time	0.35us	±150ns
T1H	1 code ,high voltage time	0.7us	±150ns
T0L	0 code , low voltage time	0.8us	±150ns
T1L	1 code ,low voltage time	0.6us	±150ns
RES	low voltage time	Above 50μs	

Sequence chart:





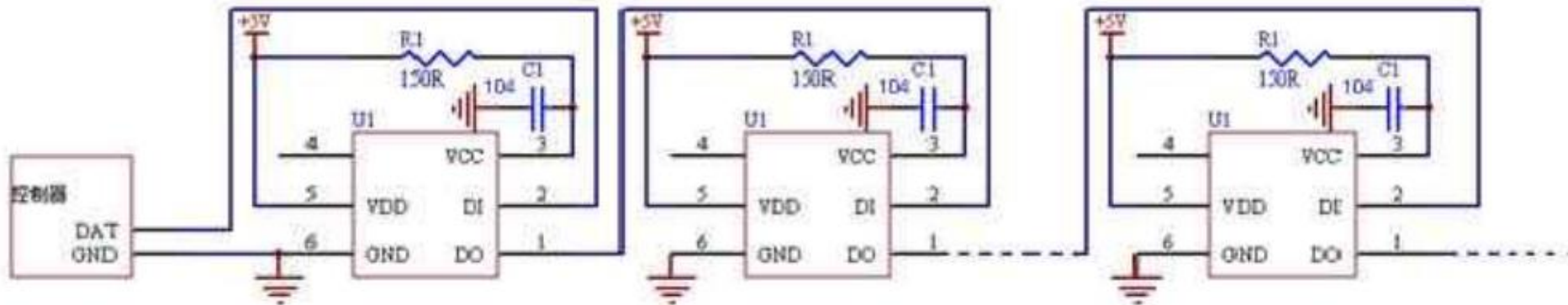
# WS2812 Timing Requirement - 2

## Composition of 24bit data:

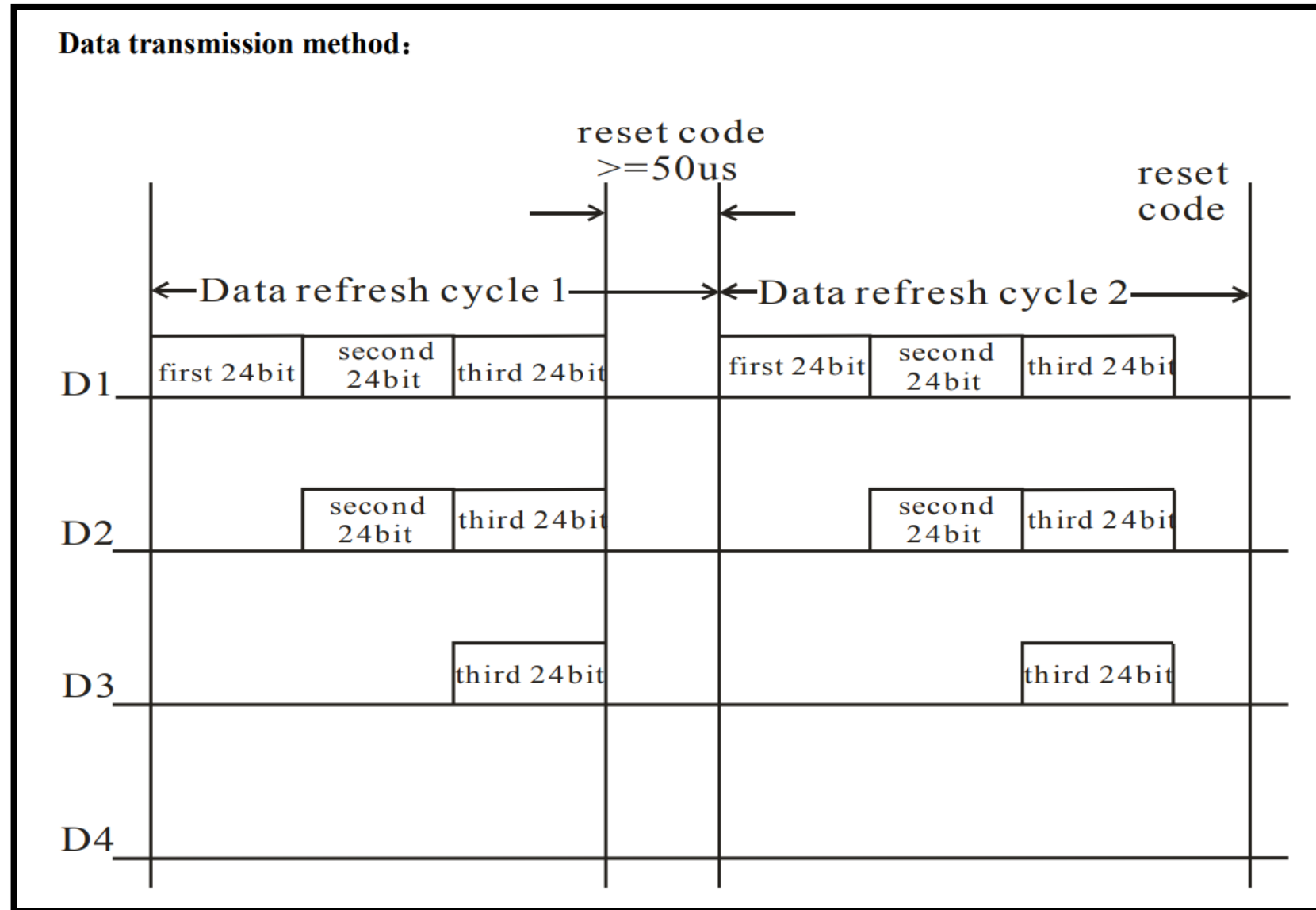
G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	B7	B6	B5	B4	B3	B2	B1	B0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Note: Follow the order of GRB to sent data and the high bit sent at first.

## Typical application circuit:



# WS2812 Timing Requirement - 3



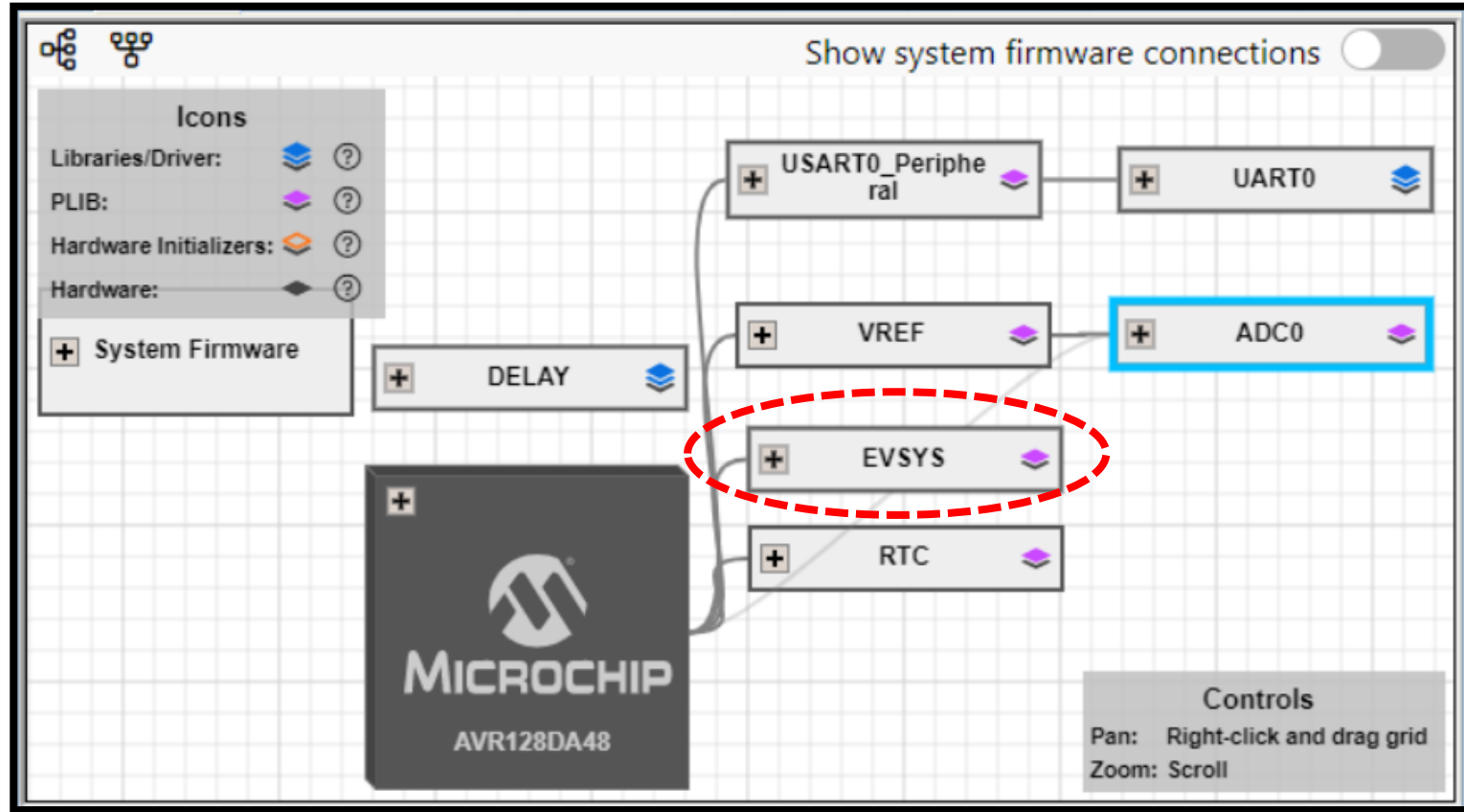
# Ex5 : AVR128DA48 使用 EVENY SYSTEM

---

Use : MPLAB XIDE V6.10

# 練習五 Project Guide

- **Project Name :** Lab\_AVR128DA48
- **Project Folder :**
  - MCHP\_ProjDev\_2023\AVR128DA48
- **使用的資源**
  - AVR128DA48 控制 LED1 ~LED4 , VR1
    - LED1 ~ LED4
    - VR1 @ PD0
  - Melody 提供的 **RTC Driver**
  - Melody 提供的 **UART Driver**
    - 115200 bps , UART @ UART0
  - Melody 提供的 **ADC Driver**
  - Melody 提供的 **EVENT SYSTEM Driver**
- **Ex5 要達成的功能**
  - 完成一個每 **200 ms** 位移一次的跑馬燈以及 **ADC 結果** 的列印
    - 位移的方向由 LED1 住次向 LED4 , 並且在盡頭轉換方向
  - 透過 **EVENT SYSTEM** , 讓 **RTC Overflow** 來觸發 **ADC0** 的轉換
  - **AIN0** 的轉換結果在 **ADC0 ISR** 被讀出



# ADC0 的 Start Event Input Enable 必須被 Enable

The screenshot displays the Microchip AVR128DA48 configuration tool interface. On the left, a grid contains various components: 'System Firmware', 'DELAY', 'USART0\_Peripheral', 'VREF', 'EVSYS', and 'RTC'. The 'VREF' component is highlighted with a blue box. A red dashed circle highlights the 'Start Event Input Enable' toggle switch in the right-hand configuration panel, which is currently turned on. The panel also shows other settings like 'Standby Mode', 'Free Running Mode', 'Initial Delay Selection', 'Sampling Delay Selection', 'Debug Run', 'Positive Input Selection', and 'Negative Input Selection'.

Icons

- Libraries/Driver: ?
- PLIB: ?
- Hardware Initializers: ?
- Hardware: ?

System Firmware

DELAY

USART0\_Peripheral

VREF

EVSYS

RTC

MICROCHIP AVR128DA48

Controls

- Pan: Right-click and drag grid
- Zoom: Scroll

Standby Mode

Free Running Mode

Initial Delay Selection: DLY0

Sampling Delay Selection: DLY0

Debug Run

Positive Input Selection: ADC input pin 0

Negative Input Selection: Ground

Start Event Input Enable



# Lets' go Ex5 by following the procedures in Video 😊

[https://youtu.be/RNWA\\_qBZ4IU](https://youtu.be/RNWA_qBZ4IU)

自動儲存 (關閉) AVR101\_AVR128DA48... • 已儲存到此電腦 搜尋 Calvin Ho - A90045

檔案 常用 插入 繪圖 設計 轉場 動畫 投影片放映 錄製 校閱 檢視 增益集 說明

剪貼簿 貼上 新 重複使用 投影片 版面配置 重設 重設 章節

字型 段落 繪圖 編輯 設計工具

115 Functional Safety

116 Development Tools

117 Ex5 : AVR128DA48 使用 EVENT SYSTEM

118 練習五 Project Guide

119 ADC0 的 Start Event Input Enable 必須 Enable

120 Let's go Ex5 by following the procedures in Video

## 練習五 Project Guide

- Project Name : Lab\_AVR128DA48
- Project Folder :
  - MCHP\_ProjDev\_2023\AVR128DA48
- 使用的資源
  - AVR128DA48 控制 LED1 ~ LED4, VR1
    - LED1 ~ LED4
    - VR1 @ PD0
  - Melody 提供的 RTC Driver
  - Melody 提供的 UART Driver
    - 115200 bps, UART @ UART0
  - Melody 提供的 ADC Driver
  - Melody 提供的 EVENT SYSTEM Driver
- Ex5 要達成的功能
  - 完成一個每 200 ms 位移一次的跑馬燈以及 ADC 結果的列印
    - 位移的方向由 LED1 往次向 LED4, 並且在盡頭轉換方向
  - 透過 EVENT SYSTEM, 讓 RTC Overflow 來觸發 ADC0 的轉換
  - AIN0 的轉換結果在 ADC0 ISR 被讀出

Microchip AVR128DA48

Microchip Proprietary and Confidential

85%

32°C 下午 05:55 2023/9/19

# Bonus : Ex6 – Configuration for TWI0

The screenshot displays the Microchip AVR128DA48 configuration tool interface. The main workspace shows a system firmware connection diagram with components like DELAY, RTC, EVSYS, USART0\_Peripheral, VREF, ADC0, TWI0\_Peripheral, and I2C0\_Host. The TWI0\_Peripheral component is highlighted with a blue border. The right-hand panel shows the configuration settings for the TWI0\_Peripheral, including Software Settings, Hardware Settings, and Interrupt Settings.

**System Firmware Connections:**

- System Firmware (Microchip AVR128DA48)
- DELAY
- RTC
- EVSYS
- USART0\_Peripheral
- VREF
- ADC0
- TWI0\_Peripheral (highlighted)
- I2C0\_Host

**Configuration Settings for TWI0\_Peripheral:**

- Easy View:** Register Initialization
- Software Settings:**
  - Interrupt Driven: ☒
- Hardware Settings:**
  - Actual Clock (Hz): 410958
- Interrupt Settings:**
  - Read Interrupt Enable: ☒
  - Write Interrupt Enable: ☒

**Controls:**

- Pan: Right-click and drag grid
- Zoom: Scroll

# Bonus : Ex6 – Configuration for I2C0\_Host

- Custom Name = I2C0 Host = API Name !

The screenshot displays the Microchip AVR128DA48 configuration tool interface. The main workspace shows a system diagram with various components connected to the AVR128DA48 microcontroller. The components include:

- System Firmware
- DELAY
- RTC
- EVSYS
- USART0\_Peripheral
- UART0
- VREF
- ADC0
- TWI0\_Peripheral
- I2C0\_Host (highlighted with a blue border)

The right-hand panel shows the configuration settings for the selected component, I2C0\_Host. The settings are organized into sections:

- Easy View** (selected tab)
- Software Settings**
  - Custom Name: I2C0\_Host (highlighted with a red dashed box)
  - Requested Speed: 400 KHz
  - Calculated Speed: 410.96 KHz
- Dependency Selector**
  - I2C Host PLIB Selector: TWI0

The bottom of the interface includes a **Controls** section with instructions: Pan: Right-click and drag grid, Zoom: Scroll.



**SMART | CONNECTED | SECURE**

***“Thank You”***